

# INTERNATIONAL STANDARD



## AMENDMENT 1

**Communication networks and systems for power utility automation –  
Part 6: Configuration description language for communication in power utility  
automation systems related to IEDs**

IECNORM.COM : Click to view the full PDF of IEC 61850-6:2009/AMD1:2018



## THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2018 IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester. If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

IEC Central Office  
3, rue de Varembe  
CH-1211 Geneva 20  
Switzerland

Tel.: +41 22 919 02 11  
[info@iec.ch](mailto:info@iec.ch)  
[www.iec.ch](http://www.iec.ch)

### About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

### About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

#### IEC Catalogue - [webstore.iec.ch/catalogue](http://webstore.iec.ch/catalogue)

The stand-alone application for consulting the entire bibliographical information on IEC International Standards, Technical Specifications, Technical Reports and other documents. Available for PC, Mac OS, Android Tablets and iPad.

#### IEC publications search - [webstore.iec.ch/advsearchform](http://webstore.iec.ch/advsearchform)

The advanced search enables to find IEC publications by a variety of criteria (reference number, text, technical committee,...). It also gives information on projects, replaced and withdrawn publications.

#### IEC Just Published - [webstore.iec.ch/justpublished](http://webstore.iec.ch/justpublished)

Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and also once a month by email.

#### Electropedia - [www.electropedia.org](http://www.electropedia.org)

The world's leading online dictionary of electronic and electrical terms containing 21 000 terms and definitions in English and French, with equivalent terms in 16 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

#### IEC Glossary - [std.iec.ch/glossary](http://std.iec.ch/glossary)

67 000 electrotechnical terminology entries in English and French extracted from the Terms and Definitions clause of IEC publications issued since 2002. Some entries have been collected from earlier publications of IEC TC 37, 77, 86 and CISPR.

#### IEC Customer Service Centre - [webstore.iec.ch/csc](http://webstore.iec.ch/csc)

If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: [sales@iec.ch](mailto:sales@iec.ch).

IECNORM.COM : Click to view the full PDF file (IEC 85050-2003/AMD1:2018)

# INTERNATIONAL STANDARD



AMENDMENT 1

**Communication networks and systems for power utility automation –  
Part 6: Configuration description language for communication in power utility  
automation systems related to IEDs**

INTERNATIONAL  
ELECTROTECHNICAL  
COMMISSION

ICS 33.200

ISBN 978-2-8322-5033-4

**Warning! Make sure that you obtained this publication from an authorized distributor.**

## 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
57/1918/FDIS	57/1940/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.

The Code Component(s) included in this IEC standard are a set of .xsd. This Code Component is published through the IEC WebSite; for details see 1.3 of the present IEC 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.

**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		

IECNORM.COM : Click to view the full PDF of IEC 61850-6:2009/AMD1:2018

## INTRODUCTION

*Add, after the first paragraph of the Introduction, the following new paragraph:*

While this part describes the language to describe the configuration of IEC 61850 systems, other parts of the standard describe how to configure the system and possible restrictions. Therefore implementations claiming conformance to this standard shall take into account constraints from the other normative references. Some references to the other parts have been included for the purpose of clarification but these references are not all inclusive.

### 1 Scope

*Replace the existing 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: 2018/01/22

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

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	2018	IEC 61850-6:2009/AMD1:2018	IEC 61850-6:2007B4
Edition 2.1	2018	IEC 61850-6:2009+AMD1:2018 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/tc57/supportdocuments/IEC\\_61850-6.2018.SCL.2007B4.full.zip](http://www.iec.ch/tc57/supportdocuments/IEC_61850-6.2018.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\_61850-6.2018.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/AMD1:2018; see the IEC 61850-6:2009/AMD1:2018 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="2018-01-22">
```

```
<Publication name="IEC_61850-6.2018_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.2007B4.txt" startingDate="2014-06-11" endingDate="2017-07-05"
startingVersion="SCL.XSD.v2007B" 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\_61850-6.2018.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.

## 2 Normative references

Add the following two new references:

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

IEC IEEE 61850-9-3, *Communication networks and systems for power utility automation – Part 9-3: Precision time protocol profile for power utility automation*

Replace the existing reference to IEC 61850-7-1 with the following new reference:

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

## 5 Intended engineering process with SCL

### 5.3 Use of SCL in the engineering process

Add, after the first sentence of the second paragraph of 5.3, the following new sentence:

Clause 10 provides more details about the engineering rights of the tool roles.

## 6 The SCL object model

### 6.1 General

Replace the existing 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 existing Figure 6 with the following new Figure 6:

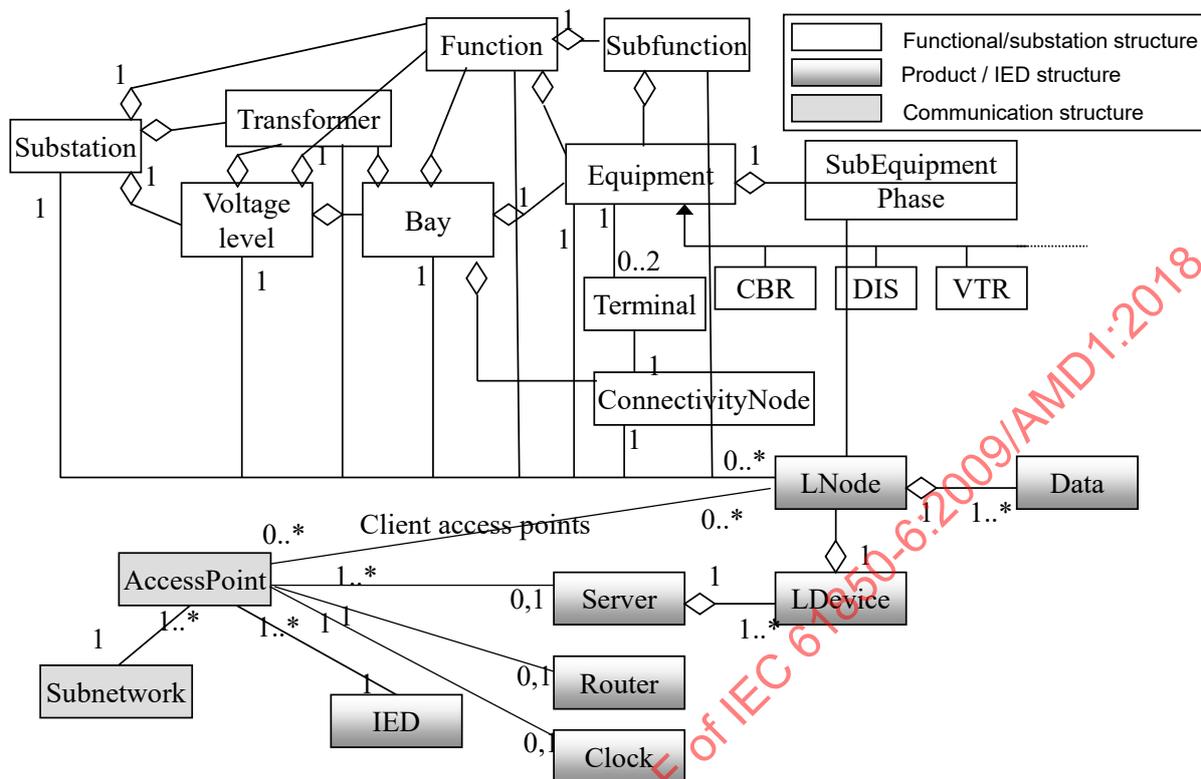


Figure 6 – SCL Substation object model

## 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 place in 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 SCL language

### 8.1 Specification method

Replace the existing paragraph following Table 1 by the following new paragraph:

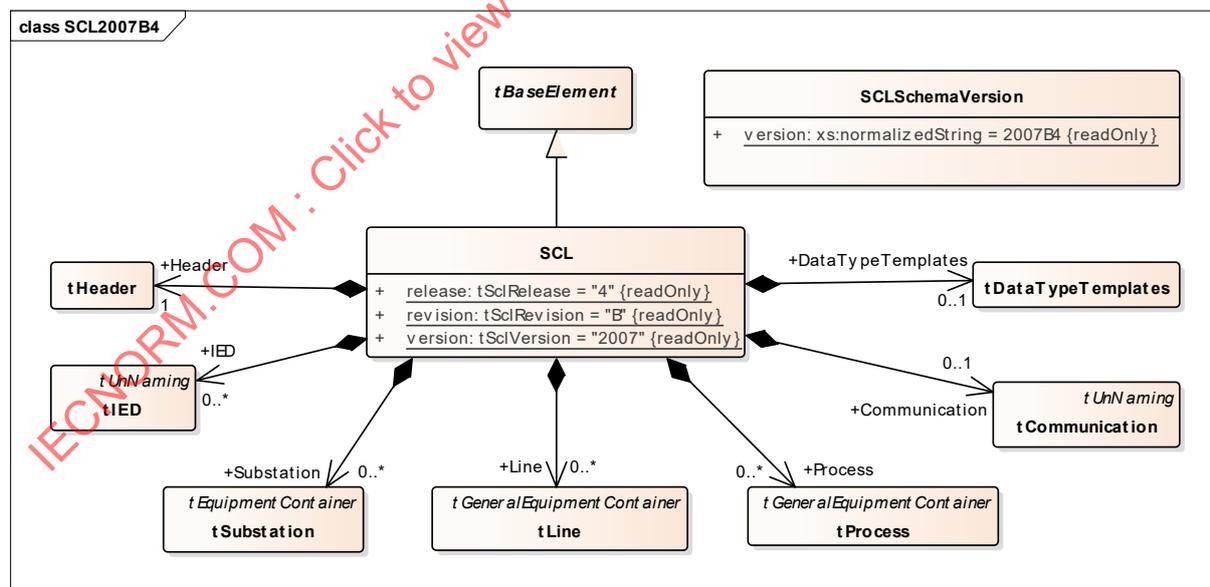
In the following schema definition clauses it is assumed that the SCL schema definition file starts as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="http://www.iec.ch/61850/2003/SCL"
  xmlns:scl="http://www.iec.ch/61850/2003/SCL"
  xmlns="http://www.iec.ch/61850/2003/SCL"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified"
  finalDefault="extension" version="9999A9">
```

where version 9999A9 is the year, index letter and release number stating the SCL schema version, which is 2007B4 for this document. The schema then ends with

```
</xs:schema>
```

Replace existing Figure 9 by the following new Figure 9:



IEC

Add, after the fourth sentence of the last paragraph of 8.1, the following new text:

SCL may use values to identify another element of the XML file. When doing this, the referenced element shall exist in the file, otherwise the structure is invalid. For example (full definition of this example is given later) in element GSE, a specific attribute called cbName indicates the element GSEControl and attribute IdInst indicates the Ldevice containing this

GSEControl, for which the GSE has been created, and if this GSEControl is not present, the GSE shall not exist.

### 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" dataSet="MyDataSet">
  <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 existing 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 *DataTemplate* 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 existing 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 The SCL syntax elements

### 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 and figure 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 of the Process and Line elements is shown as UML diagram in Figure 24.

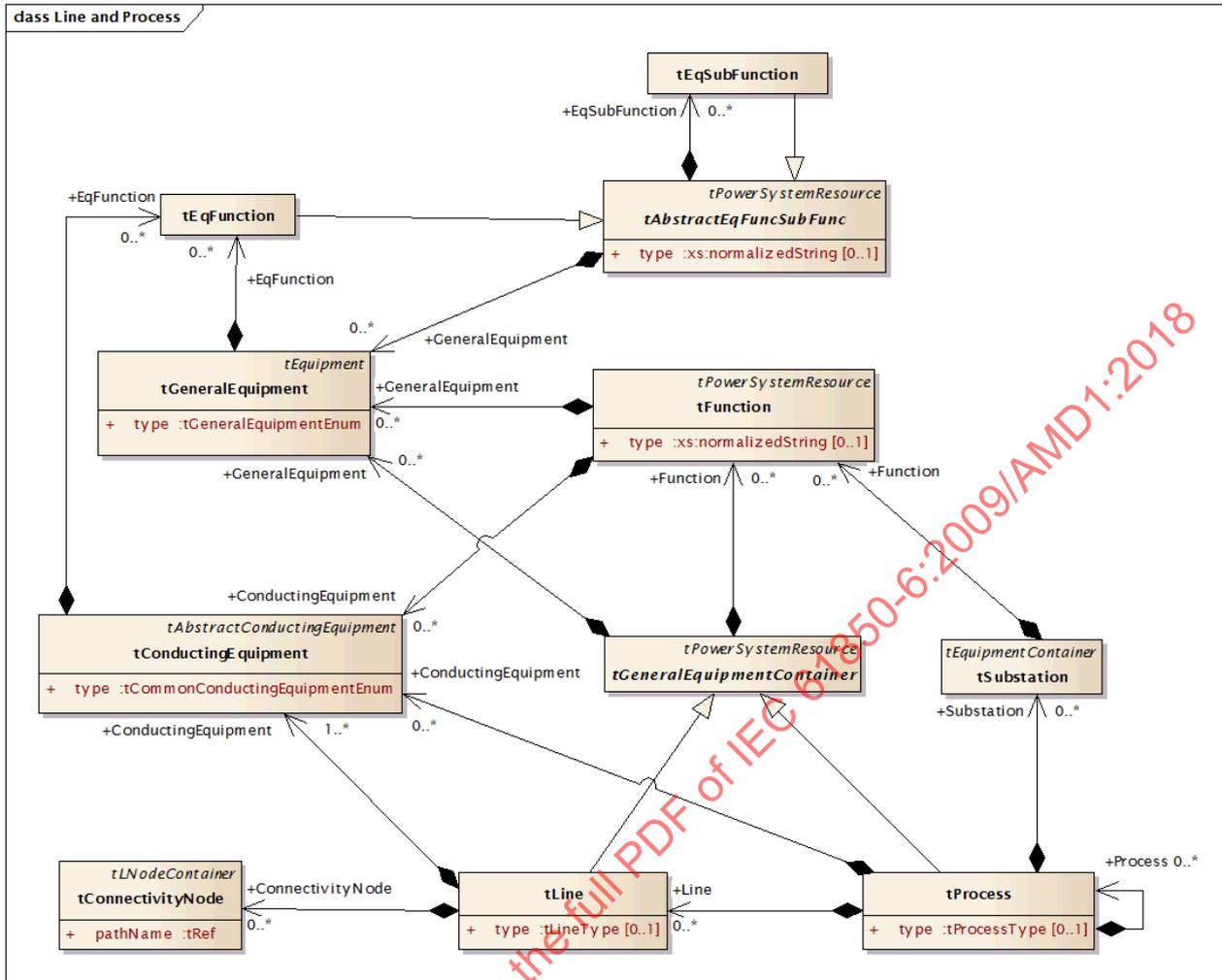


Figure 24 – UML diagram of Process and Line elements

The general *Process* element (of type *tProcess*, see Figure 24), 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 *lnInst*, *lnClass*, *iedName*, *ldInst*, 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="@jiedName"/>
            <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="@jiedName"/>
            <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 existing Figure 16 with the following new figure:

IECNORM.COM : Click to view the full PDF of IEC 61850-6:2009/AMD1:2018



```
<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:unique>
        </xs:element>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```



```

    <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="@InInst"/>
            <xs:field xpath="@InClass"/>
            <xs:field xpath="@iedName"/>
            <xs:field xpath="@IdInst"/>
            <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="@InInst"/>
            <xs:field xpath="@InClass"/>
            <xs:field xpath="@iedName"/>
            <xs:field xpath="@IdInst"/>
            <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="@InInst"/>
            <xs:field xpath="@InClass"/>
            <xs:field xpath="@iedName"/>
            <xs:field xpath="@IdInst"/>
            <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 <i>iedName</i> =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 `lnInst` should be different if several LNs with identical `lnClass` 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: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.

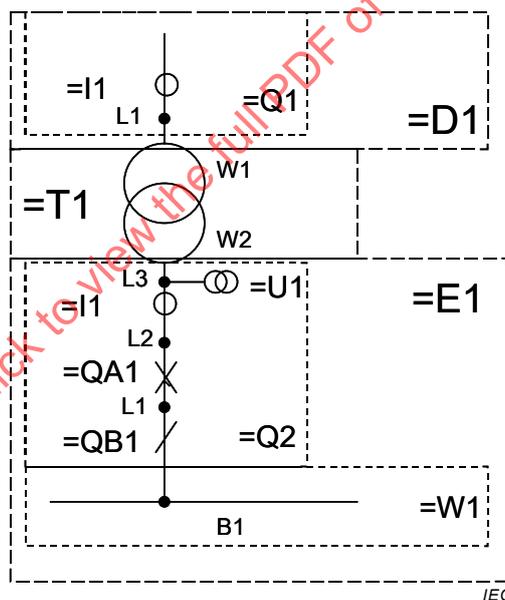
**Table 9 – General Equipment codes from IEC 61850-7-4**

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 InInst="1" InClass="PDIF" IdInst="F1"/>
      <LNode InInst="1" InClass="TCTR" IdInst="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 InInst="1" InClass="PDIS" IdInst="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 InInst="1" InClass="CILO" IdInst="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 InInst="2" InClass="CSWI" IdInst="C1"/>
          <LNode InInst="2" InClass="CILO" IdInst="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>
    </VoltageLevel>
    <Bay name="W1">
      <ConnectivityNode name="B1" pathName="baden220_132/E1/W1/B1"/>
    </Bay>
  </VoltageLevel>
</Substation>
</SCL>

```





Replace existing 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 shall deduce them from the first imported ICD or IID file for this IED type. In the case that they are missing in an SCD, SSD or SED file, the default 2003A (SCL Edition 1) applies.

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

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

Service capability	Description
ClientServices	<p>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.</p> <p>The attribute <i>supportsLdName</i> indicates that the client/subscriber understands explicit LD name setting at servers (mandatory for clients from Ed2 onwards).</p> <p>Default for an attribute missing: false</p> <p>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:</p> <p><i>maxAttributes</i>: the maximal number of data set entries (as sum across all received data sets) supported by the client.</p> <p><i>maxReports</i>: the maximal number of report control blocks the client can subscribe to.</p> <p><i>maxGOOSE</i>: the maximum number of GOOSE messages receivable.</p> <p><i>maxSMV</i>: the maximum number of SV messages receivable.</p> <p><i>rGOOSE</i>: if true, GOOSE subscription at network level (layer 3) is supported; default = false.</p> <p><i>rSV</i>: if true, SV message subscription at network level (layer 3) is supported; default = false.</p> <p><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.</p> <p>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.</p> <p>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.</p>
DynAssociation	<p>*All services for dynamic building of associations.</p> <p>The <i>max</i> attribute indicates the guaranteed number of dynamic associations which are possible under all circumstances.</p>
SettingGroups SGEdit ConfSG	<p>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.</p> <p>Both have the following attribute:</p> <p><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.</p>
GetDirectory	<p>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.</p> <p>Includes the IEC 61850-7-2 services GetServerDirectory, GetLogicalDeviceDirectory, GetLogicalNodeDirectory</p>
GetDataObjectDefinition	<p>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</p>

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	<p>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.</p> <p>The attribute meaning is:</p> <p><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</p> <p>Note that the control block settings might also restrict data set modifications even if <i>modify</i>=TRUE.</p>
DynDataSet	<p>Services to dynamically create and delete data sets. Refers to IEC 61850-7-2 services CreateDataSet and DeleteDataSet.</p> <p>The attribute meaning is:</p> <p><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</p>
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 <i>ctlModel</i> attribute. It is a service without attributes.
ConfReportControl	<p>Capability of static (by configuration via SCL) creation of report control blocks. If no value of ReportSettings is fix, the system configurator can delete and add ReportControls up to the defined max value.</p> <p>The attribute's meaning is:</p> <p><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.</p> <p><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.</p> <p><i>bufConf</i> – boolean. TRUE means, the <i>buffered</i> attribute of report control blocks can be changed via SCL by a system configurator.</p> <p><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.</p> <p>All RCBs instantiable by the system configurator are indexed.</p>
GetCBValues	Read values of control blocks. It is a service without attributes
ConfLogControl	<p>Capability of static (by configuration via SCL) creation of log control blocks. If no value of LogSettings is fix, the system configurator can delete and add LogControls up to the defined max value.</p> <p>The attribute's meaning is:</p> <p><i>max</i> – maximum number of instantiable log control blocks</p>

Service capability	Description
ReportSettings	<p>The report control block attributes for which a change of setting is possible at engineering time or online with services SetURCBValues respective SetBRCBValues:</p> <p>The attribute's meaning is:</p> <p>cbName – control block name (Fix, Conf; default = Fix)</p> <p>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)</p> <p>rptID – report identifier</p> <p>optFields – optional fields to include in report</p> <p>bufTime – buffer time</p> <p>trgOps – trigger options enable</p> <p>intgPd – integrity period</p> <p>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).</p> <p>owner – if true, the report control block possesses an owner attribute. Default is false</p>
LogSettings	<p>The log control block attributes for which a change of setting is possible at engineering time or with service SetLCBValues:</p> <p>The attribute's meaning is:</p> <p>cbName – control block name (Fix, Conf; default = Fix)</p> <p>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)</p> <p>logEna – log enable</p> <p>trgOps – trigger options</p> <p>intgPd – integrity period</p>
GSESettings	<p>The GSE control block attributes for which a change of setting is possible at engineering time or with service SetGoCBValues:</p> <p>The attribute's meaning is:</p> <p>cbName – control block name(Fix, Conf; default = Fix)</p> <p>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)</p> <p>appID – application identifier</p> <p>dataLabel – value for the object reference if the corresponding element ist being sent (applies only to GSSE control blocks, which are deprecated)</p> <p><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.</p> <p>GSESettings allows the following subelement:</p> <p>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.</p>

Service capability	Description
SMVSettings	<p>The SMV control block attributes for which a change of setting is possible at engineering time or with service SetMSVCBValues respective SetUSVCBValues:</p> <p>The attribute's meaning is:</p> <p>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)                      pdcTimeStamp – 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.</p> <p>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</p> <p>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.</p>
ConfLNs	<p>Describes what can be configured for LNs defined in an ICD file</p> <p>The attribute meanings are:</p> <p>fixPrefix – if false, prefixes can be set/changed; default = false                      fixLnInst – if false, LN instance numbers can be changed: default = false</p>
ConfLdName	<p>If this element is present, the IED allows as a server to define functional LD names (by means of the LDevice ldName attribute)</p>
GSEDir	<p>GSE directory services according to IEC 61850-7-2. This capability has no attributes.</p>
GOOSE	<p>This element shows that the IED can be a GOOSE publisher according to IEC 61850-7-2. If no value of GSESettings is fix, the system configurator can delete and add GoCBs up to the defined max value. The subscriber capability shall be exposed in ClientServices.</p> <p>The attributes meaning is:</p> <p><i>max</i> = maximum number of GOOSE control blocks, which are configurable for publishing. Shall be greater or equal to the number of defined GoCBs. (max= 0 at an access point means that this access point does not support GOOSE sending).</p> <p><i>fixedOffs</i> = true indicates that fixed offsets are supported and can be set individually per GoCB. Default: false.</p> <p><i>goose</i>: if true, layer 2 GOOSE sending is supported. Default: true.</p> <p><i>rGOOSE</i>: if true, layer 3 (network level) GOOSE sending is supported. Default: false.</p>
GSSE	<p>This element shows that the IED can be a binary data GSSE publisher or subscriber according to IEC 61850-7-2.</p> <p>The attributes meaning is:</p> <p><i>max</i> – maximum number of GSSE control blocks, which are configurable. Max=0 means only GSSE client.</p> <p>GSSE is deprecated since Edition 2.</p>

Service capability	Description
SMVsc	<p>This element shows that the IED can be a Sampled Value publisher according to IEC 61850-7-2. If no value of SMVSettings is fix, the system configurator can delete and add SVCBs up to the defined max value. The subscriber capability shall be exposed in ClientServices.</p> <p>The attributes meaning is:</p> <p><i>max</i> = maximum number of SMV control blocks, which are configurable for publishing. Shall be greater than or equal to the number of defined SVCBs. (max=0 at an access point means that this access point does not support SV sending).</p> <p><i>delivery</i>: 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.</p> <p><i>deliveryConf</i>: indicates if above type (multicast, unicast) can be configured by the system configurator; default: false.</p> <p><i>sv</i>: if true, layer 2 SV is supported; default: true.</p> <p><i>rSV</i>: if true, layer 3 (network level; multicast only) SV is supported; default false.</p>
FileHandling	<p>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.</p> <p><i>mms</i>: MMS based file handling, default=true</p> <p><i>ftp</i>: FTP based file handling, default=false</p> <p><i>ftps</i>: FTP with SSL based file handling, default = false</p>
SupSubscription	<p>This element shows the capability to supervise GOOSE or SMV subscriptions. The attribute meaning is:</p> <p><i>maxGo</i> – 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 &gt;0, at least one instance shall exist in the data model.</p> <p><i>maxSv</i> – maximum number of SV subscription supervision LNs (LN class LSVS) to be instantiated on the IED. Default: only preconfigured LNs can be used. If &gt;0, at least one instance shall exist in the data model.</p> <p>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</p>
ConfSigRef	<p>This element shows the IED capability to include input references into logical nodes. The attribute meaning is:</p> <p><i>max</i> – maximum number of input references (e.g. data objects InRef and BkRef, 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.</p>
CommProt	<p>Specifies additional lower layer protocols. Observe that IP V4 always needs to be supported, if IP is used at all.</p> <p><i>ipv6</i>: if true, IP V6 is supported. Default is false</p>
TimeSyncProt	<p>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.:</p> <p><i>sntp</i>: the SNTP protocol; default is true</p> <p><i>c37_238</i>: the IEEE 1588 method as defined in C37.238; default is false (deprecated)</p> <p><i>iec61850_9_3</i>: the IEEE 1588 method as defined in IEC 61850-9-3; default is false</p> <p><i>other</i>: any other method, e.g. PPS (although no protocol); default is false</p>

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 true, calculation of a signature is supported. Default: false. <i>encryption</i> : if true, 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		identical
	SGEdit	resvTms	identical
	ConfSG		identical
	ConfSG	resvTms	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 text and tables after the XML schema with the following new text and tables:

The LN0 contains the following elements: *GSEControl* (see 9.3.10), *SampledValueControl* (see 9.3.11), and *SettingControl* (see 9.3.12). Furthermore, it inherits *ReportControl*, *Log* and *LogControl* from the base type *tAnyLN*, as well as the *DOI* and *Inputs* element.

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.

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 16.

**Table 16 – Attributes of the LN0 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

### Restrictions

- The LN0 LN class is always LLN0, so no *inst* attribute is needed. For the referencing of links to LN0, *InInst* shall be missing, and *InClass* shall be LLN0.

The Logical Node (type *tLN*) is described as follows:

```
<xs:complexType name="tLN">
  <xs:complexContent>
    <xs:extension base="tAnyLN">
      <xs:attribute name="InClass" type="tLNClassEnum" use="required"/>
      <xs:attribute name="inst" type="tLNInst" use="required"/>
      <xs:attribute name="prefix" type="tPrefix" use="optional" default=""/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

*tAnyLN*, the super-type of both *tLN0* and *tLN*, is defined as follows:

```
<xs:complexType name="tAnyLN" abstract="true">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:sequence>
        <xs:element name="DataSet" type="tDataSet" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="ReportControl" type="tReportControl" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="LogControl" type="tLogControl" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="DOI" type="tDOI" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniqueSDI_DAlinDOI">
            <xs:selector xpath="."/"/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
        <xs:element name="Inputs" type="tInputs" minOccurs="0">
          <xs:unique name="uniqueExtRefInInputs">
            <xs:selector xpath="./scl:ExtRef"/>
            <xs:field xpath="@iedName"/>
            <xs:field xpath="@IdInst"/>
            <xs:field xpath="@prefix"/>
            <xs:field xpath="@InClass"/>
            <xs:field xpath="@InInst"/>
            <xs:field xpath="@doName"/>
            <xs:field xpath="@daName"/>
            <xs:field xpath="@intAddr"/>
          </xs:unique>
        </xs:element>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

```

        </xs:unique>
    </xs:element>
    <xs:element name="Log" type="scl:tLog" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
<xs:attribute name="InType" type="tName" use="required"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
    
```

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**

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

The optional DOI elements in an LN definition can be used to define special instance-related values for data objects and their attributes by using SDI elements for data object or attribute structure parts (if needed) and DAI elements per final attribute (see DOI definition in 0). The data objects and attributes referenced here shall however already be defined within the LNodeType definition of the LN, referenced with the LNType attribute of the LN. The DOI elements at this place for this instance shall NOT define new DOs or new attributes, which are not contained in the LNodeType. For example, the pulse length configuration parameter of a DPC CDC, specified with 100 ms in the LNodeType, is overwritten here with a value of 300 ms for this special DO. If the same value applies to several occurrences, typical values can be defined within the DataTypeTemplate section. In this case an individual value can be used to override the typical value.

**Restrictions**

- The LN Name consisting of *prefix*, *InClass* and *inst* shall be unique within the scope of the logical device, if a server is defined, or else within the scope of the IED.
- The *inst* attribute shall be a number with no more than 7 digits.
- The *prefix* follows the restrictions stated in IEC 61850-7-2.
- The *Log* element is only allowed in LLN0, and in some special LN classes explicitly defined in other parts of this standard or in other standards.

### 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 SCSM dependent.

Replace existing Table 19 with the following new table:

**Table 19 – Attributes of the DAI element**

Attribute name	Description
desc	The description text for the DAI element
name	The name of the Data attribute whose value is given. It is the last name part in a structured attribute name.
sAddr	Short address of this Data attribute
valKind	The meaning of the value from the engineering phases. If missing, the <i>valKind</i> from the type definition applies for any attached value.
ix	Index of the DAI element in case of an array type
valImport	if true, an IED / IED configurator can import values modified by another tool, even if <i>valKind</i> =RO or <i>valKind</i> =Conf; the default value is defined in the DatatypeTemplate section. It is the responsibility of the IED configurator to assure value consistency and value allowance even if <i>valImport</i> is true.

Replace the existing text following Table 19 with the following new text:

The DAI element contains a subset of the DA attributes, and shall be used within an IED DOI specification if some instance specific attribute values are set or typical attribute values overwritten. The DAI name together with the index value, if supplied, shall be unique at the specified level.

The subset of data or data attributes are described as follows:

```

<xs:complexType name="tSDI">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element name="SDI" type="tSDI"/>
        <xs:element name="DAI" type="tDAI"/>
      </xs:choice>
      <xs:attribute name="name" type="tAttributeNameEnum" use="required"/>
      <xs:attribute name="ix" type="xs:unsignedInt" use="optional"/>
      <xs:attribute name="sAddr" type="xs:normalizedString" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

The SDI element stands for a substructure name part, either from a DO (corresponding to SDO in LNodeType) or a DA substructure name, except the final (leaf) attribute name. The

SDI element contains either the elements *SDI* for a further structure name part, or *DAI* for the final attribute element with the value(s).

The attributes of the SDI element are defined as shown in Table 20.

**Table 20 – Attributes of the SDI element**

Attribute name	Description
desc	A description text for the SDI part
name	Name of the SDI (structure part)
ix	Index of the SDI element in case of an array type
sAddr	The short address for the whole structure. Shall only used if no lower level attribute has an sAddr value at instance or type definition.

Add to the bullet list after Table 20 the following bullet:

- the combination of name value and ix value shall be unique at each level (e.g. inside a DOI element)

**9.3.7 Data set definition**

Replace existing Table 22 and the text following Table 22 as follows.

**Table 22 – Attributes of the FCDA element**

Attribute name	Description
ldInst	The LD where the DO resides; shall always be specified except for GSSE
prefix	Prefix identifying together with <i>InInst</i> and <i>InClass</i> the LN where the DO resides; optional, default value is the empty string
InClass	LN class of the LN where the DO resides; shall always be specified except for GSSE DataLabel empty string
InInst	Instance number of the LN where the DO resides; shall be specified except for LLN0
doName	A name identifying the DO (within the LN). A name standardized in IEC 61850-7-4. The doName attribute is optional only if the data set is assigned to a GSSE control block. For elements or parts of structured data object types, all name parts are contained, separated by dots (.), down to (but without) the level where the fc is defined. If an SDO array element is selected, the appropriate name part shall contain at its end before a possible dot the array element number in the form ( <i>ArrayElementNumber</i> ).
daName	The attribute name – if missing, all attributes with functional characteristic given by <i>fc</i> are selected. For elements or parts of structured data types, all name parts are contained, separated by dots (.), starting at the level where the fc is defined. If an attribute's array element is selected, the appropriate attribute name part shall contain at its end before any separating dot the array element number in the form ( <i>ArrayElementNumber</i> ).
fc	All attributes of this functional constraint are selected. Possible constraint values see IEC 61850-7-2 or the <i>fc</i> definition in 9.5
ix	An index to select an array element in case that one of the data elements is an array. The ix value shall be identical to the <i>ArrayElementNumber</i> value in the doName or daName part.

The order of data within a message based on this data set definition shall be the FCDA order in the data set. If an FCDA specifies a set of attributes via *fc*, then the order of data values is specified by the data object definition and the attribute order in the corresponding LNs *DOType*. The attribute order inside the DatatypeTemplate section *DOType* and *DAType* elements shall follow the CDC definitions of IEC 61850-7-3.

## Restrictions

- If *daName* contains a non empty value, then the *fc* value must be valid for the attribute (i.e. defined identically at the appropriate *LNNodeType* definition), otherwise the SCL file processing shall be stopped with an error message.
- If all attributes of the FCDA (except *fc*) are missing or empty, then this corresponds to an empty string in a GSSE DataLabel definition (*fc* value should be ST) – in all other data sets, this is not allowed and *IdInst*, *InClass*, *InInst*, *doName* and *fc* are mandatory. Observe that GSSE is deprecated and only contained here for backwards compatibility.
- All control blocks, which reference a data set, shall be contained in the same LN as the data set definition. Therefore, the data set reference within all control blocks only contains the LN relative data set name (Name attribute at DataSet element), and not its full name (which also contains the LD name and LN name according to IEC 61850-7-2).
- The data set name length is syntactically restricted to 32 characters, which corresponds to a general name length specifiable in the Services element of 64 characters. Observe that the default name length of the Services element of 32 means that the allowed usable data set name length is even shorter, i.e. 20 characters.

## Data set example

```
<DataSet name="Example">
  <FCDA IdInst="C1" prefix="" InInst="1" InClass="CSWI" doName="Pos" fc="ST"/>
  <FCDA IdInst="C1" prefix="" InInst="2" InClass="CSWI" doName="Pos" fc="ST"/>
  <FCDA IdInst="C1" prefix="" InInst="1" InClass="MMXU" doName="A" fc="MX"/>
  <FCDA IdInst="C1" prefix="" InInst="1" InClass="MMXU" doName="PhV.phsA" fc="MX" daName="cVal" />
  <FCDA IdInst="C1" InInst="1" InClass="PVOC" doName="TmAST" fc="SP" daName="crvPts(2).xVal" ix="2"/>
  <FCDA IdInst="C1" InInst="1" InClass="MHAI" doName="HPhV.phsAHar(3)" fc="MX" daName="mag" ix="3"/>
</DataSet>
```

## Table 22 – Attributes of the FCDA element

Replace the existing text of the third sentence in the *doName* row with the following new text:

The *doName* attribute is mandatory if the dataset is used for any other service than the deprecated GSSE.

### 9.3.8 Report Control Block

Replace existing Table 23 with the following new table:

**Table 23 – Attributes of the report control block element**

Attribute name	Description
name	Name of the report control block. This name is relative to the LN hosting the RCB, and shall be unique within the LN
desc	The description text
datSet	The name of the data set to be sent by the report control block; datSet should only be missing within an ICD-File, or to indicate an unused control block. The referenced data set must be in the same LN as the control block.
intgPd	Integrity period in milliseconds – see IEC 61850-7-2. Only relevant if trigger option <i>period</i> is set to true
rptID	Identifier for the report control block, optional; if not used, its value shall be set to NULL (see IEC 61850-7-2)
confRev	The configuration revision number of this report control block. The value 0 is only allowed for a control block without data set reference. A reset by the system configurator is not allowed.
buffered	Specifies if reports are buffered or not – see IEC 61850-7-2; default: false
bufTime	Buffer time – see IEC 61850-7-2; default: 0
indexed	If true, the report control block instance names are built from the supplied name, followed by an index number from 01 up to maximum 99. Default: true. The value false is not allowed for SCT created instances.

Replace the two paragraphs before Table 24 with the following new text:

The RptEnabled element contains the list of client LNs which can enable the reporting and for which the appropriate control block instance shall be reserved.

The attributes given in Table 24 are used.

Replace the first bullet in the bullet list **ClientLn restrictions** after Table 25 as follows:

#### **ClientLn restrictions:**

- Both the *iedName* and *IdInst* (as specified in the attribute group *agLDRef*) shall be specified with non-zero length. If the reference is to an LN at a pure client access point, then the value of *IdInst* shall be LD0. Note that earlier SCL versions might use other *IdInst* values here.

Add the following bullet to the bullet list **Report Control block restrictions** after the **ClientLn** restrictions:

- The *confRev* value 0 is only allowed for newly created control blocks without data set reference. For any change of the data set structure or the data set reference, inclusive removal of this, *confRev* shall be incremented.

### 9.3.10 GSE control block

Replace existing Table 27 with the following new table:

**Table 27 – Attributes of the GSE control block element**

Attribute name	Description
name	The name identifying this GOOSE control block
desc	A description text
datSet	The name of the data set to be sent by the GSE control block. For type=GSSE, the FCDA definitions in this data set shall be interpreted as DataLabels according to IEC 61850-7-2. The attribute datSet should only be missing within an ICD-File, or to indicate an unused control block. It resides in LLN0 like the control block
confRev	The configuration revision number of this control block. It is recommended to increment this by 10 000 on each configuration change, to distinguish this from online changes leading to an increment of 1 only
type	If the <i>type</i> is GSSE ( <i>deprecated</i> ), then only single indication and double indication data types are allowed for the data items referenced in the data set, otherwise all data types are allowed. Note that on stack level, each type might be mapped differently to message formats. The default type value is GOOSE
appID	A system wide unique identification of the application to which the GOOSE message belongs; relates to the GoID of part 7-2.
fixedOffs	Default value false. If set to true it shows all receivers, that the values within the GOOSE message have fixed offset in the GOOSE message until a reconfiguration. This might mean for an MMS mapping that e.g. for integer values always the maximum size is used, although ASN.1 would allow a shorter coding.
securityEnabled	Default: None. Allows to configure the security options per control block instance: Signature or SignatureAndEncryption. Only those indicated by the McSecurity element of the GSESetting are allowed.

Add the following bullets to the bullet list **Restrictions** after Table 28:

- The confRev value 0 is only allowed for newly created control blocks without data set reference. For any change of the data set structure or the data set reference, inclusive removal of this, confRev shall be incremented.
- The type GSSE shall no longer be used. It exists only for backwards compatibility.

### 9.3.11 Sampled value control block

Replace existing Table 29 with the following new table:

**Table 29 – Attributes of the sampled value control block element**

Attribute name	Description
name	A name identifying this SMV control block
desc	The description text
datSet	The name of the data set whose values shall be sent; datSet should only be missing within an ICD-File, or to indicate an unused control block. A referenced data set must reside in LLN0.
confRev	The configuration revision number of this control block; mandatory. It is recommended to increment it by 10000 on any configuration change, to distinguish this from online configuration changes leading to an increment of 1 only
smvID	Multicast CB: the MsvID for the sampled value definition as defined in IEC 61850-7-2 Unicast CB: the UsvID as defined in IEC 61850-7-2
multicast	<i>false</i> indicates Unicast SMV services only meaning that smvID = UsvID
smpRate	Sample rate as defined in IEC 61850-7-2. If no smpMod is defined, in samples per period, else as stated by smpMod.

Attribute name	Description
nofASDU	Number of ASDU (Application service data unit) – see IEC 61850-9-2
smpMod	The sampling mode as defined in IEC 61850-7-2; default: SmpPerPeriod; if supported by the IED, also SmpPerSec and SecPerSample can be chosen. In these cases smpRate defines the appropriate sample number per second, or seconds between samples.
securityEnabled	Default: None. Allows to configure the security options per control block instance: Signature or SignatureAndEncryption. Only those indicated by the McSecurity element of the SMVSetting are allowed.

Replace existing Table 30 with the following new table:

**Table 30 – Attributes of the Smv Options element**

Attribute name	Description
refreshTime	The meaning of the options is described in IEC 61850-7-2. If any of the attributes is set to true, the appropriate values shall be included into the SMV telegram
sampleRate	
dataSet	The meaning of the options is described in IEC 61850-7-2. If the attribute is set to true, the dataset name shall be included into the SMV telegram
security	See IEC 61850-9-2 for description
synchSourceId	if true, the SV message contains the identity of the synchronizing master clock according to IEC 61850-9-3; default = false

Add the following bullet to the bullet list **Restrictions** after Table 31:

- The confRev value 0 is only allowed for newly created control blocks without data set reference. For any change of the data set structure or the data set reference, inclusive removal of this, confRev shall be incremented.

### 9.3.12 Setting control block

Replace the XML schema and the following text, including Table 32, with the following new text:

```

<xs:complexType name="tSettingControl">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:attribute name="numOfSGs" use="required">
        <xs:simpleType>
          <xs:restriction base="xs:unsignedInt">
            <xs:minInclusive value="1"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
      <xs:attribute name="actSG" use="optional" default="1">
        <xs:simpleType>
          <xs:restriction base="xs:unsignedInt">
            <xs:minInclusive value="1"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
      <xs:attribute name="resvTms" type="xs:unsignedShort" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

The attributes are identical to those of the setting group control block in IEC 61850-7-2.

The attributes of the setting control block element are defined in Table 32.

**Table 32 – Attributes of the setting control block element**

Attribute name	Description
desc	The description text
numOfSGs	The number of setting groups available. The value shall be > 0.
actSG	The number of the setting group to be activated when loading the configuration. The default value is 1. Any SCL value shall be > 0.
resvTms	The time in seconds the SGCB stays reserved for editing. After this time the IED automatically closes an edit session, if the client has not closed it or not confirmed any changes (see IEC 61850-7-2). If this function is not supported, the attribute shall be missing. The appropriate IED capability (Table 11) defines if a system tool can modify any value supplied by the IED tool.

### 9.3.13 Binding to external signals

Replace the first paragraph of 9.3.13 with the following new text:

The input section element *Inputs* defines all external signals, i.e. signals sent from other LNs mostly on other IEDs, which are needed by the LN application to fulfill its function. The section allows also the binding of the signal to an IED internal address *intAddr*.

The ExtRef elements of Input sections can principally be used for the following purposes:

- 1) Specifying expected inputs (input templates) of logical devices or logical nodes from the view of the IED configuration tool. The *intAddr* attribute shall be set to a non empty value. The *pDO* might specify via a standardized DO name the needed input CDC, additional *pDA* values might indicate the expected data type of an attribute and the *pLN* might document the expected function providing the input. *pServT* might provide the expected service type for this input value. These attribute values, if supplied by the IED configurator, shall not be modified by the system configurator. A system configurator binding to these input templates must assure that the CDC specified by *pDO* and, if given, the attribute (base) type specified by *pDA* are met. For a final engineered data flow also the *pServT*, if given, shall be met by the *serviceType* value.
- 2) Specifying expected external signal inputs to a logical device or logical node from the system engineer (system configuration tool) view. Beneath the external reference in this case the *serviceType* might be set. The *intAddr* attribute, all source control block related attributes and all *pXX* attributes are missing.
- 3) Documenting the actual incoming signals to an IED / logical device / logical node based on the configured data flow. In this case beneath the reference to the source control block also the service type is given.
- 4) Documenting the binding of incoming external signals to IED internal addresses for those signals actually used by the IED. Here the *intAddr* value after binding of the external signal is also specified.

The engineering process allows IED tools that need the system configuration tool to perform a binding of predefined IED input signals to an external signal source. In this case the IED must have the attribute *noIctBinding* of the *ClientServices* capability element set to 'true'. IED tools that are not restricted here (*noIctBinding* = 'false') may provide predefined IED input signal templates, but must accept also unbound external signals for binding by the IED tool.

Some example use cases can be found in Annex H.

Replace existing Table 33 with the following new table:

**Table 33 – Attributes of the Input/ExtRef element**

Attribute name	Description
iedName	The name of the IED from where the input comes. For IED internal references the value @ may be used.
ldInst	The LD instance name from where the input comes
prefix	The LN prefix
InClass	The LN class according to IEC 61850-7-x. Used to indicate the InClass of the concrete binding part within a SCD.
InInst	The instance id of this LN instance of above LN class in the IED; missing for a reference in LLN0. For backwards compatibility also the empty string shall be accepted for LLN0
doName	A name identifying the DO (within the LN). In case of structured DO, the name parts are concatenated by dots (.). Used to indicate the DO (within the LN) of the concrete binding part within a SCD.
daName	The attribute designating the input. The IED tool should use an empty value if it has some default binding (intAddr) for all process input attributes of a DO (fc = ST or MX), especially for t and q. If the attribute belongs to a data type structure, then the structure name parts shall be separated by dots (.)
intAddr	The internal address to which the input is bound. Only the IED tool of the concerned IED shall use the value. All other tools shall preserve it unchanged.
desc	A free description / text. Can e.g. be used at system engineering time to tell the IED engineer the purpose of this incoming data
serviceType	Optional, values: Poll, Report, GOOSE, SMV, Used to indicate the used service if the data flow is configured.
srcLDInst	The LD inst of the source control block – if missing, same as ldInst above
srcPrefix	The prefix of the LN instance, where the source control block resides; if missing, no prefix
srcLNClass	The LN class of the LN, where the source control block resides; if missing, LLN0
srcLNInst	The LN instance number of the LN where the source control block resides – if missing, no instance number exists (LLN0)
srcCBName	The source CB name; if missing, then all other srcXX attributes should also be missing, i.e. no source control block is given.
pDO	a preconfigured DO name to indicate an expected DO name and CDC. Any binding must match the CDC.
pLN	a preconfigured LN class indicating an expected LN class containing the DO indicated by pDO
pDA	a preconfigured data attribute indicating the expected attribute. If configured, any bound attribute must match the data type specified by specified pDO CDC and pDA attribute value
pServT	a preconfigured service type indicating an expected service type. if configured, serviceType must match its value.

Replace the rest of 9.3.13 after Table 33 as follows:

If attribute *intAddr* is specified and *iedName* as well as *ldInst* are missing, this allows an IED configuration tool to specify the available internal addresses for later binding to external references. In this case any values at *pDO* and possibly *pDA* allow to specify the CDC respective attribute type expected for this *intAddr*. Together with *InClass* this allows to specify a functionally expected input. The *desc* attribute allows to give it additional informal meaning, the *serviceType* attribute the intended service quality. If at least the *pDO* attribute is specified, then a missing *pDA* attribute means at least all the operational value attribute(s) of the DO, i.e. *stVal*, *mag*, *t*, *q* etc. It might allow binding of additional attributes supported by the service type. In this case, *intAddr* can also specify the addresses of all operational attributes in some IED tool specific way. So, both usages are supported additionally to a full link specification: *intAddr* with missing *iedName* and *ldInst* e.g. within an ICD or IID file, or only an external reference by means of *iedName/ldInst/prefix/InClass/InInst* but without *intAddr* coming from an

SCD file . The *desc* attribute can be used to indicate the purpose of this link, dependent on the usage above.

The system configuration tool can create and remove ExtRef definitions without an *intAddr* value and bind external references to an *intAddr* supplied via an ExtRef template from the IED tool or remove the binding from it. It is not allowed to change *intAddr* values or values of the *pXxx* attributes, nor is it allowed to remove ExtRefs with a defined *intAddr*. The IED tool can bind external references to internal addresses, and supply templates with *intAddr* values, which should contain at least additionally the *pDO* to indicate the expected CDC of an external signal. Some use cases are described in Annex H.

If the same input data can be received by the IED by different communication services (for example by report and by GOOSE), it is up to the project engineer or the IED respective its tool implementation to decide which one shall be taken, except an IED supplied ExtRef template exists which specifies already the needed *serviceType*. Any decision can be documented by means of the *srcXX* attributes, which allow to define the source control block for this input data. If *serviceType* is set to *Poll*, then no source control block shall be specified; this means that the client shall/will poll the input data by means of read requests.

Observe that the *serviceType* as well as the *srcXx* attributes are syntactically optional. For an SCD with completed data flow engineering they shall be supplied consistently with the data flow configured at the source.

Table 51 summarizes the usage of ExtRef attributes for the above defined purposes.

**Table 51 – Usage of ExtRef attributes in different use cases**

ExtRef attributes	Purpose use cases			
	ICD/IID – expected input of logical nodes (input template)	SSD – SCD – specifying expected external signal	SCD – Documenting the actual incoming signals	ICD/IID/SCD Documenting the binding of incoming external signals
iedName	na	M	M	M
IdInst	na	M	M	M
prefix	na	MD	MD	MD
InClass	O	M	M	M
InInst	na	MD	MD	MD
doName	na	M	M	M
daName	na	O	O	M
intAddr	M	na	C	M
pLN	O	na	C	C
pDO	M – indirect specification of expected type on reception in combination with optional pDA	na	C	C
pDA	O – in combination with pDO indirect specification of expected basic type on reception	na	C	C
pServT	O – in combination with pDO specification of expected service type	na	C	C
desc	M	O	O	M
serviceType	na	O	M	M
srcLDInst	na	na	MCD	MCD
srcPrefix	na	na	MCD	MCD
srcLNClass	na	na	MCD	MCD
srcLNInst	na	na	MCD	MCD
srcCBName	na	na	MC	MC
<p>Na not applicable.</p> <p>MD Mandatory; can be missing if default applies.</p> <p>MC Mandatory, if the <i>serviceType</i> value is NOT 'Poll'.</p> <p>MCD Mandatory if <i>srcCBName</i> is given and the default does not apply.</p> <p>C after SCT binding as supplied in ICD/IID file, else missing (na).</p> <p>O Optional.</p> <p>M Mandatory.</p>				

The Inputs / ExtRef feature could also be used to describe the IED internal signal connections, i.e. the external reference is an IED internal signal. It is up to the IED tool if it supports this option either as documentation or even as manufacturer independent IED internal engineering feature. If such a binding is used, the ExtRef.serviceType shall be missing.

### 9.4.2 Subnetwork definition

Replace existing Table 36 with the following new table:

**Table 36 – Attributes of the ConnectedAP element**

Attribute	Description
iedName	a name identifying the IED
apName	a name identifying this access point within the IED
desc	some descriptive text for this access point at this subnetwork
redProt	The redundancy protocol used at this access point: allowed values are hsr, prp, rstp, none; no default value, i.e. value not known if the attribute is missing. The allowed values are restricted by the IED capabilities (Services/RedProt).

Add, after Table 36, the following new text:

For further restrictions see IEC 61850-7-2:2003, Table 1.

### 9.4.6 Physical connection parameters

Replace the existing third paragraph of 9.4.6 and Table 39 with the following new text:

The type attribute specifies the type of physical connection of this access point to the bus, while the value then specifies the instance of this type (for example type="Plug", value is "ST"). The *PhysConn* type *Connection* defines a first physical connection, while the type *RedConn* can identify an additional physically redundant connection at the same access point, e.g. for redundancy protocol types PRP and HSR. Allowed types and values shall be defined in the stack mapping. The P element can be repeated with other types, if one value is not sufficient. For the physical connections defined in IEC 61850-8-1, the types and corresponding values as shown in Table 39 shall be used.

**Table 39 – PhysConn P-Type definitions**

PhysConn type	P type	Recommended values (IEC 61850-8-1-related)
Connection, RedConn	Type	<i>10BaseT</i> , <i>100BaseT</i> etc. for electrical connection <i>FOC</i> for optical connection <i>Radio</i> for radio connection, for example WLAN
	Plug	<i>RJ45</i> for electrical plug <i>ST</i> for bajonet plug (optical glass)
	Cable	The identification of a physical cable for this connection, which connects this connection point to another connection point
	Port	The identification of a port or terminal at this access point to which a cable is connected (see connection point above) or may be connected

### 9.5.1 General

Add, after the second sentence of the first paragraph of 9.5.1, the following new text:

The content of a logical node type shall follow the definitions in IEC 61850-7-3 and IEC 61850-7-4xx and possible vendor-specific extensions that shall follow the rules defined in IEC 61850-7-1:2011, Clause 14.

Add, at the end of the the first paragraph of 9.5.1, the following new text:

An SCT may be required to change the id of an LNodeType, DOType, DAType or EnumType but it is not allowed to change the underlying structure.

Add the following sentence after Figure 23:

All identifiers (XML attribute id) of all elements in the DataTypeTemplate section have a maximum allowed length of 255 characters and are restricted to xs:token without blanks.

### 9.5.2 LNodeType definitions

Replace existing Table 41 with the following new table:

**Table 41 – Attributes of the LNodeType element**

Attribute	Description
id	A reference identifying this LN type within this SCL section; used by the LN attribute LNTType of from a LNode definition in the process section to reference this definition
desc	An additional text describing this LN type
iedType	The manufacturer IED type of the IED to which this LN type belongs – deprecated
InClass	The LN base class of this type as specified in IEC 61850-7-x; observe that here an enumeration exists, which allows extensions (names containing only capital letters)

Replace existing Table 42 with the following new table:

**Table 42 – Attributes of the DO element**

Attribute	Description
name	The data object name as specified for example in IEC 61850-7-4
type	The type references the id of a DOType definition
accessControl	Access control definition for this DO. If it is missing, then any higher-level access control definition applies
transient	If set to true, it indicates that the Transient definition from IEC 61850-7-4 applies
desc	Descriptive text for the DO element

### 9.5.3 DO type definition

Replace existing Table 43 with the following new table:

**Table 43 – Attributes of the DOType element**

Attribute	Description
id	The (global) identification of this DOType. Used to reference this type.
iedType	The type of the IED to which this DOType belongs. The empty string allows references for all IED types, or from the Substation section without IED identification.
cdc	The basic CDC (Common Data Class) as defined in IEC 61850-7-3.
desc	Description of this DOType

## 9.5.4 Data attribute (DA) definition

### 9.5.4.1 General

Replace existing Table 45, and the rest of 9.5.4.1 after Table 45, with the following new text:

**Table 45 – Data type mapping**

IEC 61850-7-x basic type	XML Schema (xs) data type	Value representation
INT8, INT16, INT24, INT32, INT64 INT8U, INT16U, INT24U, INT32U	integer	An integer number, no decimal fraction (99999)
FLOAT32, FLOAT64	double	A number with or without a decimal fraction (+999.99999), or with an exponent (+9.999999e+999)
BOOLEAN	boolean	<i>false</i> , <i>true</i> or 0, 1
ENUMERATED	normalizedString	The enumeration element names as defined in the EnumType associated with the DA element as string values
OCTET STRING	base64Binary	Coding according to 6.8 of RFC 2045
VISIBLE STRING	normalizedString	A character string without tabs, linefeeds and carriage return, restricted to 8-bit characters (ISO/IEC 8859-1 limited to characters with UTF-8 single byte coding)
UNICODE STRING	normalizedString	A character string without tabs, linefeeds and carriage return. All characters in an XML file are principally Unicode, for example in UTF-8 coding
ObjectReference	normalizedString	The reference to an IEC 61850 object, as defined in IEC 61850-7-2. Additionally IED internal references especially in ICD files shall be stated as follows: @IdInst/InName[.doName[.attributename]], i.e. the @ replaces the IED name. Observe that the appropriate online values always must be absolute names considering any IdName values, i.e. the relative names are only allowed inside SCL.
Timestamp (UTC time)	dateTime	Coding without time zone, e.g. 2007-12-31T21:01:12.345
Currency	normalizedString	See IEC 61850-7-3: values are coded according to ISO 4217 3-character currency code

NOTE It is not intended to specify Quality values and CODED ENUM values in an SCL file, as these only belong to live process data. The same is true for EntryID values.

The meaning of the value for an IED configurator can be different depending on the device capabilities, the functional characteristic of the attribute, and the stage of the engineering process. The DA attribute *valKind* allows the specification of this meaning. It is ignored for all cases not specified in Table 46 (for example for the *q* and *t* attributes). The attributes *valKind* and *valImport* make no sense for DA elements with *bType* = "Struct" and shall then be ignored by the tools.

```
<xs:simpleType name="tValKindEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="Spec"/>
    <xs:enumeration value="Conf"/>
    <xs:enumeration value="RO"/>
    <xs:enumeration value="Set"/>
  </xs:restriction>
</xs:simpleType>
```

**Table 46 – Attribute value kind (Valkind) meaning**

Valkind value	Functional constraints	Engineering process stage	Meaning
Spec	Non operational (CF, DC)	Specification phase	The wanted value determined at specification phase typically in an SSD file
Conf	CF, DC, operational attribute of a CDC used for settings, EX (attribute dataNs)	IED template, after IED engineering	This value is not visible online at the IED. The IED is engineered such that this value is used
RO	Operational process state attribute	IED template	The default value for the attribute to be used if the value is fix on the IED
RO	CF, DC, operational attribute of data used for settings	IED template, after IED configuration	Read only value at an IED – can only be set at configuration time
Set	CF, DC	At/after IED configuration	A determined setting value. The value is/shall be set within the IED
Set	Operational process values (except time and quality)	At/after IED configuration (possibly RO changed to Set)	The default value for the operational attribute, e.g. for startup or simulation
Set	Operational setting value (SP, SG for all data used as setting)	At/after IED configuration	The setting value for the set point respectively parameter

This allows, for example, the definition of IED capabilities (which attributes are available, which are read only), the default values an IED is delivered with (readable, changeable, or not visible at all), or the setting values for operative (for example protection) parameters.

The additional attribute *valImport* allows to define if an IED / IED configurator can import values modified by another tool from an SCD file, even if valKind=RO or valKind=Conf. The attribute *valImport* makes no sense for DA elements with bType = "Struct" and shall then be ignored by the tools.

The syntax definition is as follows. It is based on an abstract type tAbstractDataAttribute which is reused later in attribute structure definitions.

```

<xs:complexType name="tDA">
  <xs:complexContent>
    <xs:extension base="tAbstractDataAttribute">
      <xs:sequence>
        <xs:element name="ProtNs" type="tProtNs" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attributeGroup ref="agDATrgOp"/>
      <xs:attribute name="fc" type="tFCEnum" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:attributeGroup name="agDATrgOp">
  <xs:attribute name="dchg" type="xs:boolean" use="optional" default="false"/>
  <xs:attribute name="qchg" type="xs:boolean" use="optional" default="false"/>
  <xs:attribute name="dupd" type="xs:boolean" use="optional" default="false"/>
</xs:attributeGroup>

<xs:complexType name="tAbstractDataAttribute" abstract="true">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:sequence>
        <xs:element name="Val" type="tVal" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="name" type="tAttributeNameEnum" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

```

        <xs:attribute name="sAddr" type="xs:normalizedString" use="optional"/>
        <xs:attribute name="bType" type="tBasicTypeEnum" use="required"/>
        <xs:attribute name="valKind" type="tValKindEnum" use="optional" default="Set"/>
        <xs:attribute name="valImport" type="xs:boolean" use="optional" default="false"/>
        <xs:attribute name="type" type="tAnyName" use="optional"/>
        <xs:attribute name="count" type="tDACount" use="optional" default="0"/>
    </xs:extension>
</xs:complexContent>
</xs:complexType>
    
```

The attributes of the DA element are defined in Table 47.

**Table 47 – Attributes of the DA element**

Attribute	Description
desc	Some descriptive text for the attribute
name	The attribute name; the type tAttributeNameEnum restricts to the attribute names from IEC 61850-7-3, plus new ones starting with lower case letters
fc	The functional constraint for this attribute; <i>fc</i> =SE always also implies <i>fc</i> =SG; <i>fc</i> =SG means that the values are visible, but not editable
dchg, qchg, dupd	Defines which trigger option is supported by the attribute (value true means supported). One of those allowed according to IEC61850-7-3 shall be chosen.
sAddr	An optional short address of this attribute (see 9.5.4.3)
bType	The basic type of the attribute, taken from tBasicTypeEnum (see 9.5.4.2)
type	Shall only be used if <i>bType</i> = Enum or <i>bType</i> = Struct to refer to the appropriate enumeration type or DAType (attribute structure) definition
count	Optional. Shall state the number of array elements or reference the attribute stating this number in case that this attribute is an array. A referenced attribute shall exist in the same type definition. The default value 0 states that the attribute is no array.
valKind	Determines how the value shall be interpreted if any is given – see Table 46
valImport	if true, an IED / IED configurator can import values modified by another tool from an SCD file, even if valKind=RQ or valKind=Conf. It is the responsibility of the IED configurator to assure value consistency and value allowance even if valImport is true.

The attributes *name*, *fc*, and *bType* shall always be defined. All instantiable attributes contained within a DO shall be defined.

In case that the DA element has a basic type and belongs to a specific stack mapping, it shall contain the ProtNS element, which defines the stack mapping(s) to which it belongs.

### 9.5.4.2 Attribute basic types

Replace the XML schema at the start of 9.5.4.2 by the following new schema:

```

<xs:simpleType name="tPredefinedBasicTypeEnum">
    <xs:restriction base="xs:Name">
        <xs:enumeration value="BOOLEAN"/>
        <xs:enumeration value="INT8"/>
        <xs:enumeration value="INT16"/>
        <xs:enumeration value="INT24"/>
        <xs:enumeration value="INT32"/>
        <xs:enumeration value="INT64"/>
        <xs:enumeration value="INT128"/>
        <xs:enumeration value="INT8U"/>
        <xs:enumeration value="INT16U"/>
        <xs:enumeration value="INT24U"/>
        <xs:enumeration value="INT32U"/>
        <xs:enumeration value="FLOAT32"/>
    </xs:restriction>
</xs:simpleType>
    
```

```

<xs:enumeration value="FLOAT64"/>
<xs:enumeration value="Enum"/>
<xs:enumeration value="Dbpos"/>
<xs:enumeration value="Tcmd"/>
<xs:enumeration value="Quality"/>
<xs:enumeration value="Timestamp"/>
<xs:enumeration value="VisString32"/>
<xs:enumeration value="VisString64"/>
<xs:enumeration value="VisString65"/>
<xs:enumeration value="VisString129"/>
<xs:enumeration value="VisString255"/>
<xs:enumeration value="Octet64"/>
<xs:enumeration value="Unicode255"/>
<xs:enumeration value="Struct"/>
<xs:enumeration value="EntryTime"/>
<xs:enumeration value="Check"/>
<xs:enumeration value="ObjRef"/>
<xs:enumeration value="Currency"/>
<xs:enumeration value="PhyComAddr"/>
<xs:enumeration value="TrgOps"/>
<xs:enumeration value="OptFlds"/>
<xs:enumeration value="SvOptFlds"/>
<xs:enumeration value="LogOptFlds"/>
<xs:enumeration value="EntryID"/>
<!-- for 61850-8-1 Edition 2.1 -->
<xs:enumeration value="Octet6"/>
<xs:enumeration value="Octet16"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="tBasicTypeEnum">
  <xs:restriction base="tPredefinedBasicTypeEnum"/>
</xs:simpleType>

```

Replace the existing text of NOTE 1 with the following new text:

NOTE 1 INT128 and INT24U exist only for backwards compatibility reasons, and are deprecated.

### 9.5.6 Enumeration types

Replace the text after the XML schema until Table 49 by the following new text:

Enumeration definitions within a SCD file are valid for all IEDs; they are not IED type-dependent. Enumeration definitions are given in parts 7-2, 7-3 and 7-4 and other domain specific standards. Observe that these definitions may be enhanced by new versions of the standards. If private extensions of these enumerations are used as allowed by the standards, or private enumerations are defined, the name must indicate this appropriately, i.e. none of the standard defined names should be used. This is especially important for extensions, because different manufacturers might use different extensions. It is also important, if only a subrange of the enumeration value set is supported. The supported subset shall be indicated within an ICD file by an IED private enumeration type, where the unsupported values are missing.

If the semantics of the same LN class code and same data object name code for an enumeration in another IEC name space is redefined, then the enumeration type and its values shall also be kept unchanged (possibly with redefined semantics or with value extensions).

The meaning of the attributes of the EnumType element is as shown in Table 49.

### 10.1 IED configurator

*Replace the second, third and fourth paragraphs of 10.1 by the following new text:*

How an ICD file is created depends on the IED capabilities and the tool design. There is a big range available from a fix ICD file for use in each project (the only possible adaptations are the IED name or LD name and IED address), up to ICD file generation for a specific usage of the IED e.g. for a project specific bay type after extensive preengineering by means of the IED tool – see also Clause 7 – or a preengineered project specific IED based on the current state of the system (IID file generated based on a SCD file). An ICD file shall indicate the capabilities of a possibly preengineered IED. Therefore any enumeration values which are not supported by the IED shall be removed from the referenced enumeration types, and vendor or even IED type specific enumeration type names used. The full standardized enumeration definition for configuration data and settings shall only be used

- for the enumerations of SI units and multiplier
- if all enumeration values are supported
- if the data attribute supports only a single fix value (indicated by valKind="RO" & valImport="false"), and the value is supplied in the SCL file.

Any change in data model, parameter and configuration values shall be reflected in appropriate version indicators within the LN0 NamPIt data object as values within the SCL file. For predefined data sets and control blocks the version information has to be managed as defined in IEC 61850-7-2 for the confRev parameter of the control blocks, and also contained in the SCL file.

If an SCD file is imported, an IED tool may update the version and related value information and change parameter and configuration values as well as binding external data to internal signals. It may add new control blocks and data sets. It may modify preconfigured data sets (i.e. those originally created by it) if these are not used by other IEDs in some configured data flow. It is not allowed to modify data sets which are used by some other IED. The result is transferred back to the system configurator by means of an IID file. Only the following data model changes are allowed (but not mandatory):

- addition of logical devices, logical nodes or DATA within logical nodes;
- removal of logical devices, logical nodes or DATA, which are not referenced by some client / subscriber or bound to the primary system description (substation section).

*Replace the existing text of the NOTE with the following new text:*

NOTE IED online change of data flow is described and defined in IEC 61850-7-2. Online changes of data models on an IED outside SCL are outside the scope of this standard. To keep system consistency for those IEDs already integrated into a system, they could either be prevented or follow the above offline engineering rules, especially what concerns the provision of new data model revision information.

### 10.2 System configurator

*Add the following new text before the last bullet in the bullet list:*

- Binding of input sections to IED supplied input section templates.

### 10.3 Right transfer between projects

*Replace the third paragraph and the bullet list after the third paragraph with the following new text:*

The importing project can use all received information as needed (e.g. use one of the fix IEDs as client / subscriber for its own IEDs, or add its own IEDs as clients / subscribers to provided free control block instances). It has the following additional engineering rights (modification rights) only on those IEDs exported with *dataflow* right:

- addition of data sets and control blocks as allowed by the IED capabilities. Observe that the IED owner is allowed to restrict these capabilities in the SED file;
- modification of data sets allocated purely to clients in the importing project
- adding of data set members to data sets allocated to clients in both projects.
- addition of data expected to be received from its own IEDs to the Input sections of imported IEDs.
- Binding of data sent from its own IEDs to input signal templates in imported IEDs.

Add the following new text and table at the end of 10.3:

Table 52 gives an overview of allowed actions for IEDs exported with fix or data flow engineering rights.

**Table 52 – Allowed SCT engineering actions**

Exported right:	Fix		Dataflow	
	Exporting project (right stays full)	Importing project	Exporting project (right becomes Fix)	Importing project
Add clients / subscribers to existing (free) control blocks	X (conflict possible for reporting)	X	Not for exported RCBs!	X
Use as destination IED for own IEDs as senders	X	X	X	X
Add data sets and control blocks as allowed by IED capabilities	X			X
Modify data sets (delete / rearrange FCDA)	X (conflict possible)			only if all clients in own project
Add FCDA to existing data sets	X (conflict possible)			X
Add data to be received from own IEDs to the Input sections of exported IEDs.	X			X
Bind data sent from own IEDs to input signal templates of exported IEDs	X			X

IECNORM.COM : Click to view the PDF of IEC 61850-6:2009/AMD1:2018

## Annex A (normative)

### SCL syntax: XML schema definition

Replace Subclauses A.1 to A.6 with the following new text:

#### A.1 Base types

This Annex A contains the XML schema definitions for the SCL language. Observe that in case that faults need to be fixed new schema files will be provided by IEC as separate code component.

File **SCL\_BaseSimpleTypes.xsd**

#### <CODE BEGINS>

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns="http://www.iec.ch/61850/2003/SCL"
targetNamespace="http://www.iec.ch/61850/2003/SCL" elementFormDefault="qualified" attributeFormDefault="unqualified"
version="2007B4">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      SCL schema version "2007" revision "B" release 4, for IEC 61850-6 Ed. 2.1.
    </xs:documentation>
    <xs:documentation>
      COPYRIGHT (c) IEC, 2016 All rights reserved. Disclaimer: The IEC disclaims liability for any personal injury,
      property or other damages of any nature whatsoever, whether special, indirect, consequential or compensatory, directly or
      indirectly resulting from this software and the document upon which its methods are based, use of, or reliance upon.
    </xs:documentation>
  </xs:annotation>
  <xs:simpleType name="tConnectivityNodeReference">
    <xs:restriction base="xs:normalizedString">
      <xs:pattern value=".+/.+(/.)+*" />
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tAnyName">
    <xs:restriction base="xs:normalizedString" />
  </xs:simpleType>
  <xs:simpleType name="tName">
    <xs:restriction base="tAnyName">
      <xs:minLength value="1" />
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tID">
    <xs:restriction base="xs:token">
      <xs:minLength value="1" />
      <xs:maxLength value="255" />
      <xs:pattern value="\S+" />
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tAcsiName">
    <xs:restriction base="xs:Name">
      <xs:pattern value="[A-Za-z][0-9A-Za-z_]*" />
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tRestrName1stU">
    <xs:restriction base="xs:Name">
      <xs:pattern value="[A-Z][0-9A-Za-z]*" />
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tRestrName1stL">
    <xs:restriction base="xs:Name">
      <xs:pattern value="[a-z][0-9A-Za-z]*" />
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tPAddr">
    <xs:restriction base="xs:normalizedString">
      <xs:minLength value="1" />
    </xs:restriction>
  </xs:simpleType>
```

```
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="tScIVersion">
  <xs:restriction base="tName">
    <xs:pattern value="2[0-2][0-9]{2}"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tScIRevision">
  <xs:restriction base="xs:Name">
    <xs:pattern value="[A-Z]"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tScIRelease">
  <xs:restriction base="xs:unsignedByte">
    <xs:minExclusive value="0"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tEmpty">
  <xs:restriction base="xs:normalizedString">
    <xs:maxLength value="0"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tIEDName">
  <xs:restriction base="tAcSIName">
    <xs:maxLength value="64"/>
    <xs:pattern value="[A-Za-z][0-9A-Za-z_]{0,2}"/>
    <xs:pattern value="[A-Za-z][0-9A-Za-z_]{4,63}"/>
    <xs:pattern value="[A-MO-Za-z][0-9A-Za-z_]{3}"/>
    <xs:pattern value="N[0-9A-Za-z_]{2}"/>
    <xs:pattern value="No[0-9A-Za-z_]{0-9A-Za-z_}"/>
    <xs:pattern value="Non[0-9A-Za-z_]"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tIEDNameIsNone">
  <xs:restriction base="tAcSIName">
    <xs:pattern value="None"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tIEDNameOrNone">
  <xs:union memberTypes="tIEDName tIEDNameIsNone"/>
</xs:simpleType>
<xs:simpleType name="tOnlyRelativeIEDName">
  <xs:restriction base="xs:normalizedString">
    <xs:pattern value="&#x0040,"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tIEDNameOrRelative">
  <xs:union memberTypes="tIEDName tOnlyRelativeIEDName"/>
</xs:simpleType>
<xs:simpleType name="tLDName">
  <xs:restriction base="xs:normalizedString">
    <xs:maxLength value="64"/>
    <xs:pattern value="[A-Za-z][0-9A-Za-z_]*"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tLDInst">
  <xs:restriction base="xs:normalizedString">
    <xs:maxLength value="64"/>
    <xs:pattern value="[A-Za-z0-9][0-9A-Za-z_]*"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tLDInstOrEmpty">
  <xs:union memberTypes="tLDInst tEmpty"/>
</xs:simpleType>
<xs:simpleType name="tPrefix">
  <xs:restriction base="xs:normalizedString">
    <xs:maxLength value="11"/>
    <xs:pattern value="[A-Za-z][0-9A-Za-z_]*"/>
    <xs:pattern value=""/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tLNInst">
  <xs:restriction base="xs:normalizedString">
```



```
<xs:pattern value="[0-9]{1,12}"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="tLNInstOrEmpty">
  <xs:union memberTypes="tLNInst tEmpty"/>
</xs:simpleType>
<xs:simpleType name="tDataName">
  <xs:restriction base="tRestrName1stU">
    <xs:maxLength value="12"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDataSetName">
  <xs:restriction base="tAcsiName">
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tCBName">
  <xs:restriction base="tAcsiName">
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tLogName">
  <xs:restriction base="tAcsiName">
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tAccessPointName">
  <xs:restriction base="xs:normalizedString">
    <xs:maxLength value="32"/>
    <xs:pattern value="[A-Za-z0-9][0-9A-Za-z_]*"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tAssociationID">
  <xs:restriction base="xs:normalizedString">
    <xs:minLength value="1"/>
    <xs:pattern value="[0-9A-Za-z]"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tVisibleBasicLatin">
  <xs:restriction base="xs:normalizedString">
    <xs:pattern value="[&#x0020;-&#x007E;]"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tMessageID">
  <xs:restriction base="tVisibleBasicLatin">
    <xs:minLength value="1"/>
    <xs:maxLength value="129"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tFullAttributeName">
  <xs:restriction base="xs:normalizedString">
    <xs:pattern value="[a-zA-Z][a-zA-Z0-9]*\([([0-9]+\))?\([a-zA-Z][a-zA-Z0-9]*\([([0-9]+\))?)?"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tFullDOName">
  <xs:restriction base="xs:normalizedString">
    <xs:pattern value="[A-Z][0-9A-Za-z]{0,11}\([a-z][0-9A-Za-z]*\([([0-9]+\))?)?"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tSubDataName">
  <xs:restriction base="tRestrName1stL">
    <xs:minLength value="1"/>
    <xs:maxLength value="60"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tNamespaceName">
  <xs:restriction base="xs:normalizedString">
    <xs:pattern value="[&#x0020;-&#x007E;]+\:20\d\d[A-Z]"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tLineType">
  <xs:restriction base="xs:normalizedString">
    <xs:minLength value="1"/>
  </xs:restriction>
</xs:simpleType>
```



```
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="tProcessType">
  <xs:restriction base="xs:normalizedString">
    <xs:minLength value="1"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tProcessName">
  <xs:restriction base="xs:normalizedString">
    <xs:pattern value=".(/.+)*"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tEnumStringValue">
  <xs:restriction base="xs:normalizedString">
    <xs:maxLength value="127"/>
    <xs:pattern value="[\p{IsBasicLatin}\p{IsLatin-1Supplement}]*"/>
  </xs:restriction>
</xs:simpleType>
</xs:schema>
```

**<CODE ENDS>**

IECNORM.COM : Click to view the full PDF of IEC 61850-6:2009/AMD1:2018

File **SCL\_Enums.xsd**

## &lt;CODE BEGINS&gt;

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:scl="http://www.iec.ch/61850/2003/SCL" xmlns="http://www.iec.ch/61850/2003/SCL"
xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="http://www.iec.ch/61850/2003/SCL"
elementFormDefault="qualified" attributeFormDefault="unqualified" version="2007B4">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      SCL schema version "2007" revision "B" release 4, for IEC 61850-6 Ed. 2.1.

      COPYRIGHT (c) IEC, 2016. All rights reserved. Disclaimer: The IEC disclaims liability for any
      personal injury, property or other damages of any nature whatsoever, whether special, indirect, consequential or compensatory,
      directly or indirectly resulting from this software and the document upon which its methods are based, use of, or reliance upon.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="SCL_BaseSimpleTypes.xsd"/>
  <xs:simpleType name="tPredefinedPTypeEnum">
    <xs:restriction base="xs:Name">
      <xs:enumeration value="IP"/>
      <xs:enumeration value="IP-SUBNET"/>
      <xs:enumeration value="IP-GATEWAY"/>
      <xs:enumeration value="OSI-NSAP"/>
      <xs:enumeration value="OSI-TSEL"/>
      <xs:enumeration value="OSI-SSEL"/>
      <xs:enumeration value="OSI-PSEL"/>
      <xs:enumeration value="OSI-AP-Title"/>
      <xs:enumeration value="OSI-AP-Invoke"/>
      <xs:enumeration value="OSI-AE-Qualifier"/>
      <xs:enumeration value="OSI-AE-Invoke"/>
      <xs:enumeration value="MAC-Address"/>
      <xs:enumeration value="APPID"/>
      <xs:enumeration value="VLAN-PRIORITY"/>
      <xs:enumeration value="VLAN-ID"/>
      <xs:enumeration value="SNTP-Port"/>
      <xs:enumeration value="MMS-Port"/>
      <xs:enumeration value="DNSName"/>
      <xs:enumeration value="IPv6FlowLabel"/>
      <xs:enumeration value="IPv6ClassOfTraffic"/>
      <xs:enumeration value="C37-118-IP-Port"/>
      <xs:enumeration value="IP-UDP-PORT"/>
      <xs:enumeration value="IP-TCP-PORT"/>
      <xs:enumeration value="IPv6"/>
      <xs:enumeration value="IPv6-SUBNET"/>
      <xs:enumeration value="IPv6-GATEWAY"/>
      <xs:enumeration value="IPv6-IGMPv3Src"/>
      <xs:enumeration value="IP-IGMPv3Src"/>
      <xs:enumeration value="IP-ClassOfTraffic"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tExtensionPTypeEnum">
    <xs:restriction base="xs:normalizedString">
      <xs:pattern value="[A-Z][0-9A-Za-z-]*"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tPTypeEnum">
    <xs:union memberTypes="tPredefinedPTypeEnum tExtensionPTypeEnum"/>
  </xs:simpleType>
  <xs:simpleType name="tPredefinedPTypeEnumPhysConnEnum">
    <xs:restriction base="xs:Name">
      <xs:enumeration value="Type"/>
      <xs:enumeration value="Plug"/>
      <xs:enumeration value="Cable"/>
      <xs:enumeration value="Port"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tPTypeEnumPhysConnEnum">
    <xs:union memberTypes="tPredefinedPTypeEnumPhysConnEnum tExtensionPTypeEnum"/>
  </xs:simpleType>
  <xs:simpleType name="tPredefinedAttributeNameEnum">
    <xs:restriction base="xs:Name">
      <xs:enumeration value="T"/>
      <xs:enumeration value="Test"/>
      <xs:enumeration value="Check"/>
      <xs:enumeration value="SIUnit"/>
    </xs:restriction>
  </xs:simpleType>

```

```

<xs:enumeration value="Oper"/>
<xs:enumeration value="SBO"/>
<xs:enumeration value="SBOw"/>
<xs:enumeration value="Cancel"/>
<xs:enumeration value="Addr"/>
<xs:enumeration value="PRIORITY"/>
<xs:enumeration value="VID"/>
<xs:enumeration value="APPID"/>
<xs:enumeration value="TransportInUse"/>
<xs:enumeration value="IPClassOfTraffic"/>
<xs:enumeration value="IPv6FlowLabel"/>
<xs:enumeration value="IPAddressLength"/>
<xs:enumeration value="IPAddress"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="tExtensionAttributeNameEnum">
  <xs:restriction base="tRestrName1stL">
    <xs:maxLength value="60"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tAttributeNameEnum">
  <xs:union memberTypes="tPredefinedAttributeNameEnum tExtensionAttributeNameEnum"/>
</xs:simpleType>
<xs:simpleType name="tPredefinedCommonConductingEquipmentEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="CBR"/>
    <xs:enumeration value="DIS"/>
    <xs:enumeration value="VTR"/>
    <xs:enumeration value="CTR"/>
    <xs:enumeration value="GEN"/>
    <xs:enumeration value="CAP"/>
    <xs:enumeration value="REA"/>
    <xs:enumeration value="CON"/>
    <xs:enumeration value="MOT"/>
    <xs:enumeration value="EFN"/>
    <xs:enumeration value="PSH"/>
    <xs:enumeration value="BAT"/>
    <xs:enumeration value="BSH"/>
    <xs:enumeration value="CAB"/>
    <xs:enumeration value="GIL"/>
    <xs:enumeration value="LIN"/>
    <xs:enumeration value="RES"/>
    <xs:enumeration value="RRC"/>
    <xs:enumeration value="SAR"/>
    <xs:enumeration value="TCF"/>
    <xs:enumeration value="TCR"/>
    <xs:enumeration value="IFL"/>
    <xs:enumeration value="FAN"/>
    <xs:enumeration value="SCR"/>
    <xs:enumeration value="SMC"/>
    <xs:enumeration value="PMP"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tExtensionEquipmentEnum">
  <xs:restriction base="xs:Name">
    <xs:minLength value="3"/>
    <xs:pattern value="E[A-Z]*"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tCommonConductingEquipmentEnum">
  <xs:union memberTypes="tPredefinedCommonConductingEquipmentEnum tExtensionEquipmentEnum"/>
</xs:simpleType>
<xs:simpleType name="tPowerTransformerEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="PTR"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tTransformerWindingEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="PTW"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tPredefinedGeneralEquipmentEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="AXN"/>
    <xs:enumeration value="BAT"/>
    <xs:enumeration value="MOT"/>
  </xs:restriction>

```



```
<xs:enumeration value="FAN"/>
<xs:enumeration value="FIL"/>
<xs:enumeration value="PMP"/>
<xs:enumeration value="TNK"/>
<xs:enumeration value="VLV"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="tExtensionGeneralEquipmentEnum">
  <xs:restriction base="xs:Name">
    <xs:minLength value="3"/>
    <xs:pattern value="E[A-Z]*"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tGeneralEquipmentEnum">
  <xs:union memberTypes="tPredefinedGeneralEquipmentEnum tExtensionGeneralEquipmentEnum"/>
</xs:simpleType>
<xs:simpleType name="tServiceSettingsNoDynEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="Conf"/>
    <xs:enumeration value="Fix"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tServiceSettingsEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="Dyn"/>
    <xs:enumeration value="Conf"/>
    <xs:enumeration value="Fix"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tRedProtEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="none"/>
    <xs:enumeration value="hsr"/>
    <xs:enumeration value="prp"/>
    <xs:enumeration value="rstp"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tSMVDeliveryEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="unicast"/>
    <xs:enumeration value="multicast"/>
    <xs:enumeration value="both"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tPhaseEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="A"/>
    <xs:enumeration value="B"/>
    <xs:enumeration value="C"/>
    <xs:enumeration value="N"/>
    <xs:enumeration value="all"/>
    <xs:enumeration value="none"/>
    <xs:enumeration value="AB"/>
    <xs:enumeration value="BC"/>
    <xs:enumeration value="CA"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tAuthenticationEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="none"/>
    <xs:enumeration value="password"/>
    <xs:enumeration value="weak"/>
    <xs:enumeration value="strong"/>
    <xs:enumeration value="certificate"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tAssociationKindEnum">
  <xs:restriction base="xs:token">
    <xs:enumeration value="pre-established"/>
    <xs:enumeration value="predefined"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tLPHDEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="LPHD"/>
  </xs:restriction>
</xs:simpleType>
```



```
<xs:simpleType name="tLLN0Enum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="LLN0"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tSystemLNGroupEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="L[A-Z]*"/>
    <xs:pattern value="LLN0"/>
    <xs:enumeration value="LLN0"/>
    <xs:enumeration value="LPHD"/>
    <xs:enumeration value="LCCH"/>
    <xs:enumeration value="LGOS"/>
    <xs:enumeration value="LSVS"/>
    <xs:enumeration value="LTIM"/>
    <xs:enumeration value="LTMS"/>
    <xs:enumeration value="LTRK"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupAEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="A[A-Z]*"/>
    <xs:enumeration value="ANCR"/>
    <xs:enumeration value="ARCO"/>
    <xs:enumeration value="ARIS"/>
    <xs:enumeration value="ATCC"/>
    <xs:enumeration value="AVCO"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupCEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="C[A-Z]*"/>
    <xs:enumeration value="CALH"/>
    <xs:enumeration value="CCGR"/>
    <xs:enumeration value="CILO"/>
    <xs:enumeration value="CPOW"/>
    <xs:enumeration value="CSWI"/>
    <xs:enumeration value="CSYN"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupFEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="F[A-Z]*"/>
    <xs:enumeration value="FCNT"/>
    <xs:enumeration value="FCSD"/>
    <xs:enumeration value="FFIL"/>
    <xs:enumeration value="FLIM"/>
    <xs:enumeration value="FPID"/>
    <xs:enumeration value="FRMP"/>
    <xs:enumeration value="FSPT"/>
    <xs:enumeration value="FXOT"/>
    <xs:enumeration value="FXUT"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupGEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="G[A-Z]*"/>
    <xs:enumeration value="GAPC"/>
    <xs:enumeration value="GGIO"/>
    <xs:enumeration value="GLOG"/>
    <xs:enumeration value="GSAL"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupIEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="I[A-Z]*"/>
    <xs:enumeration value="IARC"/>
    <xs:enumeration value="IHMI"/>
    <xs:enumeration value="ISAF"/>
    <xs:enumeration value="ITCI"/>
    <xs:enumeration value="ITMI"/>
  </xs:restriction>
</xs:simpleType>
```



```
        <xs:enumeration value="ITPC"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="tDomainLNGroupKEnum">
      <xs:restriction base="xs:Name">
        <xs:length value="4"/>
        <xs:pattern value="K[A-Z]*"/>
        <xs:enumeration value="KFAN"/>
        <xs:enumeration value="KFIL"/>
        <xs:enumeration value="KPMP"/>
        <xs:enumeration value="KTNK"/>
        <xs:enumeration value="KVLV"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="tDomainLNGroupMEnum">
      <xs:restriction base="xs:Name">
        <xs:length value="4"/>
        <xs:pattern value="M[A-Z]*"/>
        <xs:enumeration value="MDIF"/>
        <xs:enumeration value="MENV"/>
        <xs:enumeration value="MFLK"/>
        <xs:enumeration value="MHAI"/>
        <xs:enumeration value="MHAN"/>
        <xs:enumeration value="MHYD"/>
        <xs:enumeration value="MMDC"/>
        <xs:enumeration value="MMET"/>
        <xs:enumeration value="MMTN"/>
        <xs:enumeration value="MMTR"/>
        <xs:enumeration value="MMXN"/>
        <xs:enumeration value="MMXU"/>
        <xs:enumeration value="MSQI"/>
        <xs:enumeration value="MSTA"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="tDomainLNGroupPEnum">
      <xs:restriction base="xs:Name">
        <xs:length value="4"/>
        <xs:pattern value="P[A-Z]*"/>
        <xs:enumeration value="PDIF"/>
        <xs:enumeration value="PDJR"/>
        <xs:enumeration value="PDIS"/>
        <xs:enumeration value="PDOP"/>
        <xs:enumeration value="PDUP"/>
        <xs:enumeration value="PFRC"/>
        <xs:enumeration value="PHAR"/>
        <xs:enumeration value="PHIZ"/>
        <xs:enumeration value="PIOC"/>
        <xs:enumeration value="PMRI"/>
        <xs:enumeration value="PMSS"/>
        <xs:enumeration value="POPF"/>
        <xs:enumeration value="PPAM"/>
        <xs:enumeration value="PRTR"/>
        <xs:enumeration value="PSCH"/>
        <xs:enumeration value="PSDE"/>
        <xs:enumeration value="PTEF"/>
        <xs:enumeration value="PTHF"/>
        <xs:enumeration value="PTOC"/>
        <xs:enumeration value="PTOF"/>
        <xs:enumeration value="PTOV"/>
        <xs:enumeration value="PTRC"/>
        <xs:enumeration value="PTTR"/>
        <xs:enumeration value="PTUC"/>
        <xs:enumeration value="PTUF"/>
        <xs:enumeration value="PTUV"/>
        <xs:enumeration value="PUPF"/>
        <xs:enumeration value="PVOC"/>
        <xs:enumeration value="PVPH"/>
        <xs:enumeration value="PZSU"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="tDomainLNGroupQEnum">
      <xs:restriction base="xs:Name">
        <xs:length value="4"/>
        <xs:pattern value="Q[A-Z]*"/>
        <xs:enumeration value="QFVR"/>
        <xs:enumeration value="QITR"/>
        <xs:enumeration value="QIUB"/>
      </xs:restriction>
    </xs:simpleType>
```

IECNORM.COM  
Copyright © IEC 2018  
PDF of IEC 61850-6:2009/AMD1:2018

```

        <xs:enumeration value="QVTR"/>
        <xs:enumeration value="QVUB"/>
        <xs:enumeration value="QVVR"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupREnum">
    <xs:restriction base="xs:Name">
        <xs:length value="4"/>
        <xs:pattern value="R[A-Z]*/>
        <xs:enumeration value="RADR"/>
        <xs:enumeration value="RBDR"/>
        <xs:enumeration value="RBRF"/>
        <xs:enumeration value="RDIR"/>
        <xs:enumeration value="RDRE"/>
        <xs:enumeration value="RDRS"/>
        <xs:enumeration value="RFLO"/>
        <xs:enumeration value="RMXU"/>
        <xs:enumeration value="RPSB"/>
        <xs:enumeration value="RREC"/>
        <xs:enumeration value="RSYN"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupSEnum">
    <xs:restriction base="xs:Name">
        <xs:length value="4"/>
        <xs:pattern value="S[A-Z]*/>
        <xs:enumeration value="SARC"/>
        <xs:enumeration value="SCBR"/>
        <xs:enumeration value="SIMG"/>
        <xs:enumeration value="SIML"/>
        <xs:enumeration value="SLTC"/>
        <xs:enumeration value="SOPM"/>
        <xs:enumeration value="SPDC"/>
        <xs:enumeration value="SPTR"/>
        <xs:enumeration value="SSWI"/>
        <xs:enumeration value="STMP"/>
        <xs:enumeration value="SVBR"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupTEnum">
    <xs:restriction base="xs:Name">
        <xs:length value="4"/>
        <xs:pattern value="T[A-Z]*/>
        <xs:enumeration value="TANG"/>
        <xs:enumeration value="TAXD"/>
        <xs:enumeration value="TCTR"/>
        <xs:enumeration value="TDST"/>
        <xs:enumeration value="TFLW"/>
        <xs:enumeration value="TFRQ"/>
        <xs:enumeration value="TGSN"/>
        <xs:enumeration value="THUM"/>
        <xs:enumeration value="TLVL"/>
        <xs:enumeration value="TMGF"/>
        <xs:enumeration value="TMVM"/>
        <xs:enumeration value="TPOS"/>
        <xs:enumeration value="TPRS"/>
        <xs:enumeration value="TRTN"/>
        <xs:enumeration value="TSND"/>
        <xs:enumeration value="TTMP"/>
        <xs:enumeration value="TTNS"/>
        <xs:enumeration value="TVBR"/>
        <xs:enumeration value="TVTR"/>
        <xs:enumeration value="TWPH"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupXEnum">
    <xs:restriction base="xs:Name">
        <xs:length value="4"/>
        <xs:pattern value="X[A-Z]*/>
        <xs:enumeration value="XCBR"/>
        <xs:enumeration value="XSWI"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupYEnum">
    <xs:restriction base="xs:Name">
        <xs:length value="4"/>
        <xs:pattern value="Y[A-Z]*/>

```

IECNORM.COM © IEC 61850-6:2009/AMD1:2018

```
<xs:enumeration value="YEFN"/>
<xs:enumeration value="YLTC"/>
<xs:enumeration value="YPSH"/>
<xs:enumeration value="YPTR"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupZEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="Z[A-Z]**"/>
    <xs:enumeration value="ZAXN"/>
    <xs:enumeration value="ZBAT"/>
    <xs:enumeration value="ZBSH"/>
    <xs:enumeration value="ZCAB"/>
    <xs:enumeration value="ZCAP"/>
    <xs:enumeration value="ZCON"/>
    <xs:enumeration value="ZGEN"/>
    <xs:enumeration value="ZGIL"/>
    <xs:enumeration value="ZLIN"/>
    <xs:enumeration value="ZMOT"/>
    <xs:enumeration value="ZREA"/>
    <xs:enumeration value="ZRES"/>
    <xs:enumeration value="ZRRC"/>
    <xs:enumeration value="ZSAR"/>
    <xs:enumeration value="ZSCR"/>
    <xs:enumeration value="ZSMC"/>
    <xs:enumeration value="ZTCF"/>
    <xs:enumeration value="ZTCR"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNEnum">
  <xs:union memberTypes="tDomainLNGroupAEnum tDomainLNGroupCEnum tDomainLNGroupFEnum
tDomainLNGroupGEnum tDomainLNGroupIEnum tDomainLNGroupKEnum tDomainLNGroupMEnum tDomainLNGroupPEnum
tDomainLNGroupQEnum tDomainLNGroupREnum tDomainLNGroupSEnum tDomainLNGroupTEnum tDomainLNGroupXEnum
tDomainLNGroupYEnum tDomainLNGroupZEnum"/>
</xs:simpleType>
<xs:simpleType name="tPredefinedLNClassEnum">
  <xs:union memberTypes="tSystemLNGroupEnum tDomainLNEnum"/>
</xs:simpleType>
<xs:simpleType name="tExtensionLNClassEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="[A-Z]+"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tLNClassEnum">
  <xs:union memberTypes="tPredefinedLNClassEnum tExtensionLNClassEnum"/>
</xs:simpleType>
<xs:simpleType name="tPredefinedCDCEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="SPS"/>
    <xs:enumeration value="DPS"/>
    <xs:enumeration value="INS"/>
    <xs:enumeration value="ENS"/>
    <xs:enumeration value="ACT"/>
    <xs:enumeration value="ACD"/>
    <xs:enumeration value="SEC"/>
    <xs:enumeration value="BCR"/>
    <xs:enumeration value="HST"/>
    <xs:enumeration value="VSS"/>
    <xs:enumeration value="MV"/>
    <xs:enumeration value="CMV"/>
    <xs:enumeration value="SAV"/>
    <xs:enumeration value="WYE"/>
    <xs:enumeration value="DEL"/>
    <xs:enumeration value="SEQ"/>
    <xs:enumeration value="HMV"/>
    <xs:enumeration value="HWYE"/>
    <xs:enumeration value="HDEL"/>
    <xs:enumeration value="SPC"/>
    <xs:enumeration value="DPC"/>
    <xs:enumeration value="INC"/>
    <xs:enumeration value="ENC"/>
    <xs:enumeration value="BSC"/>
    <xs:enumeration value="ISC"/>
    <xs:enumeration value="APC"/>
    <xs:enumeration value="BAC"/>
  </xs:restriction>
</xs:simpleType>
```



```

<xs:enumeration value="SPG"/>
<xs:enumeration value="ING"/>
<xs:enumeration value="ENG"/>
<xs:enumeration value="ORG"/>
<xs:enumeration value="TSG"/>
<xs:enumeration value="CUG"/>
<xs:enumeration value="VSG"/>
<xs:enumeration value="ASG"/>
<xs:enumeration value="CURVE"/>
<xs:enumeration value="CSG"/>
<xs:enumeration value="DPL"/>
<xs:enumeration value="LPL"/>
<xs:enumeration value="CSD"/>
<xs:enumeration value="CST"/>
<xs:enumeration value="BTS"/>
<xs:enumeration value="UTS"/>
<xs:enumeration value="LTS"/>
<xs:enumeration value="GTS"/>
<xs:enumeration value="MTS"/>
<xs:enumeration value="NTS"/>
<xs:enumeration value="STS"/>
<xs:enumeration value="CTS"/>
<xs:enumeration value="OTS"/>
<xs:enumeration value="VSD"/>
<xs:enumeration value="ORS"/>
<xs:enumeration value="TCS"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="tExtensionCDCEnum">
  <xs:restriction base="xs:Name">
    <xs:minLength value="1"/>
    <xs:maxLength value="5"/>
    <xs:pattern value="[A-Za-z]"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tCDCEnum">
  <xs:restriction base="tPredefinedCDCEnum"/>
</xs:simpleType>
<xs:simpleType name="tFCEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="ST"/>
    <xs:enumeration value="MX"/>
    <xs:enumeration value="CO"/>
    <xs:enumeration value="SP"/>
    <xs:enumeration value="SG"/>
    <xs:enumeration value="SE"/>
    <xs:enumeration value="SV"/>
    <xs:enumeration value="CF"/>
    <xs:enumeration value="DC"/>
    <xs:enumeration value="EX"/>
    <xs:enumeration value="SR"/>
    <xs:enumeration value="BL"/>
    <xs:enumeration value="OR"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tPredefinedBasicTypeEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="BOOLEAN"/>
    <xs:enumeration value="INT8"/>
    <xs:enumeration value="INT16"/>
    <xs:enumeration value="INT24"/>
    <xs:enumeration value="INT32"/>
    <xs:enumeration value="INT64"/>
    <xs:enumeration value="INT128"/>
    <xs:enumeration value="INT8U"/>
    <xs:enumeration value="INT16U"/>
    <xs:enumeration value="INT24U"/>
    <xs:enumeration value="INT32U"/>
    <xs:enumeration value="FLOAT32"/>
    <xs:enumeration value="FLOAT64"/>
    <xs:enumeration value="Enum"/>
    <xs:enumeration value="Dbpos"/>
    <xs:enumeration value="Tcmd"/>
    <xs:enumeration value="Quality"/>
    <xs:enumeration value="Timestamp"/>
    <xs:enumeration value="VisString32"/>
    <xs:enumeration value="VisString64"/>
  </xs:restriction>
</xs:simpleType>

```



```
<xs:enumeration value="VisString65"/>
<xs:enumeration value="VisString129"/>
<xs:enumeration value="VisString255"/>
<xs:enumeration value="Octet64"/>
<xs:enumeration value="Unicode255"/>
<xs:enumeration value="Struct"/>
<xs:enumeration value="EntryTime"/>
<xs:enumeration value="Check"/>
<xs:enumeration value="ObjRef"/>
<xs:enumeration value="Currency"/>
<xs:enumeration value="PhyComAddr"/>
<xs:enumeration value="TrgOps"/>
<xs:enumeration value="OptFlds"/>
<xs:enumeration value="SvOptFlds"/>
<xs:enumeration value="LogOptFlds"/>
<xs:enumeration value="EntryID"/>
<xs:enumeration value="Octet6"/>
<xs:enumeration value="Octet16"/>
<!-- for 61850-8-1 Edition 2.1 -->
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="tBasicTypeEnum">
  <xs:restriction base="tPredefinedBasicTypeEnum"/>
</xs:simpleType>
<xs:simpleType name="tValKindEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="Spec"/>
    <xs:enumeration value="Conf"/>
    <xs:enumeration value="RO"/>
    <xs:enumeration value="Set"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tGSEControlTypeEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="GSSE"/>
    <xs:enumeration value="GOOSE"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tUnitMultiplierEnum">
  <xs:restriction base="xs:normalizedString">
    <xs:enumeration value=""/>
    <xs:enumeration value="m"/>
    <xs:enumeration value="k"/>
    <xs:enumeration value="M"/>
    <xs:enumeration value="mu"/>
    <xs:enumeration value="y"/>
    <xs:enumeration value="z"/>
    <xs:enumeration value="a"/>
    <xs:enumeration value="f"/>
    <xs:enumeration value="p"/>
    <xs:enumeration value="n"/>
    <xs:enumeration value="c"/>
    <xs:enumeration value="d"/>
    <xs:enumeration value="da"/>
    <xs:enumeration value="h"/>
    <xs:enumeration value="G"/>
    <xs:enumeration value="T"/>
    <xs:enumeration value="P"/>
    <xs:enumeration value="E"/>
    <xs:enumeration value="Z"/>
    <xs:enumeration value="Y"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tRightEnum">
  <xs:restriction base="xs:normalizedString">
    <xs:enumeration value="full"/>
    <xs:enumeration value="fix"/>
    <xs:enumeration value="dataflow"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tSDOCount">
  <xs:union memberTypes="xs:unsignedInt tRestrName1stL"/>
</xs:simpleType>
<xs:simpleType name="tDACount">
  <xs:union memberTypes="xs:unsignedInt tAttributeNameEnum"/>
</xs:simpleType>
<xs:simpleType name="tSmpMod">
```

IECNORM.COM Copyright © IEC 2018. Full PDF of IEC 61850-6:2009/AMD1:2018

```
<xs:restriction base="xs:normalizedString">
  <xs:enumeration value="SmpPerPeriod"/>
  <xs:enumeration value="SmpPerSec"/>
  <xs:enumeration value="SecPerSmp"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="tPredefinedPhysConnTypeEnum">
  <xs:restriction base="xs:normalizedString">
    <xs:enumeration value="Connection"/>
    <xs:enumeration value="RedConn"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tExtensionPhysConnTypeEnum">
  <xs:restriction base="xs:normalizedString">
    <xs:pattern value="[A-Z][0-9A-Za-z-]*"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tPhysConnTypeEnum">
  <xs:union memberTypes="tPredefinedPhysConnTypeEnum tExtensionPhysConnTypeEnum"/>
</xs:simpleType>
<xs:simpleType name="tServiceType">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="Poll"/>
    <xs:enumeration value="Report"/>
    <xs:enumeration value="GOOSE"/>
    <xs:enumeration value="SMV"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tPredefinedTypeOfSecurityEnum">
  <xs:restriction base="xs:normalizedString">
    <xs:enumeration value="None"/>
    <xs:enumeration value="Signature"/>
    <xs:enumeration value="SignatureAndEncryption"/>
  </xs:restriction>
</xs:simpleType>
</xs:schema>
```

<CODE ENDS>

IECNORM.COM : Click to view the full PDF of IEC 61850-6:2009/AMD1:2018

File **SCL\_BaseTypes.xsd**

## &lt;CODE BEGINS&gt;

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:scl="http://www.iec.ch/61850/2003/SCL" xmlns="http://www.iec.ch/61850/2003/SCL"
xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="http://www.iec.ch/61850/2003/SCL"
elementFormDefault="qualified" attributeFormDefault="unqualified" version="2007B4">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      SCL schema version "2007" revision "B" release 4, for IEC 61850-6 Ed. 2.1.

      COPYRIGHT (c) IEC, 2016. All rights reserved. Disclaimer: The IEC disclaims liability for any
      personal injury, property or other damages of any nature whatsoever, whether special, indirect, consequential or compensatory,
      directly or indirectly resulting from this software and the document upon which its methods are based, use of, or reliance upon.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="SCL_Enums.xsd"/>
  <xs:attributeGroup name="agDesc">
    <xs:attribute name="desc" type="xs:normalizedString" use="optional" default=""/>
  </xs:attributeGroup>
  <xs:complexType name="tBaseElement" abstract="true">
    <xs:sequence>
      <xs:any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
      <xs:element name="Text" type="tText" minOccurs="0"/>
      <xs:element name="Private" type="tPrivate" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:anyAttribute namespace="##other" processContents="lax"/>
  </xs:complexType>
  <xs:complexType name="tUnNaming" abstract="true">
    <xs:complexContent>
      <xs:extension base="tBaseElement">
        <xs:attributeGroup ref="agDesc"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tNaming" abstract="true">
    <xs:complexContent>
      <xs:extension base="tBaseElement">
        <xs:attribute name="name" type="tName" use="required"/>
        <xs:attributeGroup ref="agDesc"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tIDNaming" abstract="true">
    <xs:complexContent>
      <xs:extension base="tBaseElement">
        <xs:attribute name="id" type="tID" use="required"/>
        <xs:attributeGroup ref="agDesc"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tAnyContentFromOtherNamespace" abstract="true" mixed="true">
    <xs:sequence minOccurs="0" maxOccurs="unbounded">
      <xs:any namespace="##other" processContents="lax"/>
    </xs:sequence>
    <xs:anyAttribute namespace="##other" processContents="lax"/>
  </xs:complexType>
  <xs:complexType name="tText" mixed="true">
    <xs:complexContent>
      <xs:extension base="tAnyContentFromOtherNamespace">
        <xs:attribute name="source" type="xs:anyURI" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tPrivate" mixed="true">
    <xs:complexContent>
      <xs:extension base="tAnyContentFromOtherNamespace">
        <xs:attribute name="type" use="required">
          <xs:simpleType>
            <xs:restriction base="xs:normalizedString">
              <xs:minLength value="1"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="source" type="xs:anyURI" use="optional"/>
  </xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tHeader">
  <xs:sequence>
    <xs:element name="Text" type="tText" minOccurs="0"/>
    <xs:element name="History" minOccurs="0">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="Hitem" type="tHitem"
maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
  <xs:attribute name="id" type="xs:normalizedString" use="required"/>
  <xs:attribute name="version" type="xs:normalizedString" use="optional"/>
  <xs:attribute name="revision" type="xs:normalizedString" use="optional" default="" />
  <xs:attribute name="toolID" type="xs:normalizedString" use="optional"/>
  <xs:attribute name="nameStructure" use="optional" default="IEDName">
    <xs:simpleType>
      <xs:restriction base="xs:Name">
        <xs:enumeration value="IEDName"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:complexType>
<xs:complexType name="tHitem" mixed="true">
  <xs:complexContent>
    <xs:extension base="tAnyContentFromOtherNamespace">
      <xs:attribute name="version" type="xs:normalizedString" use="required"/>
      <xs:attribute name="revision" type="xs:normalizedString" use="required"/>
      <xs:attribute name="when" type="xs:normalizedString" use="required"/>
      <xs:attribute name="who" type="xs:normalizedString"/>
      <xs:attribute name="what" type="xs:normalizedString"/>
      <xs:attribute name="why" type="xs:normalizedString"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tVal">
  <xs:simpleContent>
    <xs:extension base="xs:normalizedString">
      <xs:attribute name="sGroup" type="xs:unsignedInt" use="optional"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tValueWithUnit">
  <xs:simpleContent>
    <xs:extension base="xs:decimal">
      <xs:attribute name="unit" type="xs:token" use="required"/>
      <xs:attribute name="multiplier" type="tUnitMultiplierEnum" use="optional"
default="" />
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tVoltage">
  <xs:simpleContent>
    <xs:restriction base="tValueWithUnit">
      <xs:attribute name="unit" type="xs:token" use="required" fixed="V"/>
      <xs:attribute name="multiplier" type="tUnitMultiplierEnum" use="optional"
default="" />
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tDurationInSec">
  <xs:simpleContent>
    <xs:restriction base="tValueWithUnit">
      <xs:attribute name="unit" type="xs:token" use="required" fixed="s"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>

```



```

                                <xs:attribute name="multiplier" type="tUnitMultiplierEnum" use="optional"
default=""/>
                                </xs:restriction>
                                </xs:simpleContent>
                                </xs:complexType>
                                <xs:complexType name="tDurationInMilliSec">
                                <xs:simpleContent>
                                <xs:extension base="xs:decimal">
                                <xs:attribute name="unit" type="xs:token" use="optional" fixed="s"/>
                                <xs:attribute name="multiplier" type="tUnitMultiplierEnum" use="optional"
fixed="m"/>
                                </xs:extension>
                                </xs:simpleContent>
                                </xs:complexType>
                                <xs:complexType name="tBitRateInMbPerSec">
                                <xs:simpleContent>
                                <xs:extension base="xs:decimal">
                                <xs:attribute name="unit" type="xs:normalizedString" use="optional" fixed="b/s"/>
                                <xs:attribute name="multiplier" type="tUnitMultiplierEnum" use="optional"
fixed="M"/>
                                </xs:extension>
                                </xs:simpleContent>
                                </xs:complexType>
</xs:schema>
```

**<CODE ENDS>**

IECNORM.COM : Click to view the full PDF of IEC 61850-6:2009/AMD1:2018

## A.2 Substation syntax

File **SCL\_Substation.xsd**

### <CODE BEGINS>

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:scl="http://www.iec.ch/61850/2003/SCL" xmlns="http://www.iec.ch/61850/2003/SCL"
xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="http://www.iec.ch/61850/2003/SCL"
elementFormDefault="qualified" attributeFormDefault="unqualified" version="2007B4">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      SCL schema version "2007" revision "B" release 4, for IEC 61850-6 Ed. 2.1.

      COPYRIGHT (c) IEC, 2016. All rights reserved. Disclaimer: The IEC disclaims liability for any personal injury,
      property or other damages of any nature whatsoever, whether special, indirect, consequential or compensatory, directly or
      indirectly resulting from this software and the document upon which its methods are based, use of, or reliance upon.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="SCL_BaseTypes.xsd"/>
  <xs:attributeGroup name="agVirtual">
    <xs:attribute name="virtual" type="xs:boolean" use="optional" default="false"/>
  </xs:attributeGroup>
  <xs:complexType name="tLNodeContainer" abstract="true">
    <xs:complexContent>
      <xs:extension base="tNaming">
        <xs:sequence>
          <xs:element name="LNode" type="tLNode" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tPowerSystemResource" abstract="true">
    <xs:complexContent>
      <xs:extension base="tLNodeContainer"/>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tEquipmentContainer" abstract="true">
    <xs:complexContent>
      <xs:extension base="tPowerSystemResource">
        <xs:sequence>
          <xs:element name="PowerTransformer" type="tPowerTransformer" minOccurs="0"
maxOccurs="unbounded">
            <xs:unique name="uniqueLNodeInPowerTransformer">
              <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="uniqueChildNameInPTR">
              <xs:selector xpath="/*"/>
              <xs:field xpath="@name"/>
            </xs:unique>
          </xs:element>
          <xs:element name="GeneralEquipment" type="tGeneralEquipment" minOccurs="0"
maxOccurs="unbounded">
            <xs:unique name="uniqueLNodeInGeneralEquipment">
              <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="uniqueChildNameInGE">
              <xs:selector xpath="/*"/>
              <xs:field xpath="@name"/>
            </xs:unique>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<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: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:extension>
  </xs:complexContent>

```

```

        </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="@InInst"/>
                        <xs:field xpath="@InClass"/>
                        <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="@InInst"/>
                        <xs:field xpath="@InClass"/>
                        <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="@InInst"/>
                        <xs:field xpath="@InClass"/>
                        <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="@InInst"/>
                        <xs:field xpath="@InClass"/>
                        <xs:field xpath="@iedName"/>
                        <xs:field xpath="@ldInst"/>
                        <xs:field xpath="@prefix"/>
                    </xs:unique>
                    <xs:unique name="uniqueChildNameInLTC">
                        <xs:selector xpath="/*"/>
                    </xs:unique>
                </xs:element>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>

```



```

</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>
<xs:complexType name="tSubstation">
  <xs:complexContent>
    <xs:extension base="tEquipmentContainer">
      <xs:sequence>
        <xs:element name="VoltageLevel" type="tVoltageLevel" maxOccurs="unbounded">
          <xs:unique name="uniqueChildNameInVoltageLevel">
            <xs:selector xpath="./*/>
            <xs:field xpath="@name"/>
          </xs:unique>
          <xs:unique name="uniqueLNodeInVoltageLevel">
            <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:element>
        <xs:element name="Function" type="tFunction" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInFunctionSS">
            <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="uniqueChildNameInSubstationFunc">
            <xs:selector xpath="./*/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<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="@iedName"/>
            <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="@iedName"/>
            <xs:field xpath="@ldInst"/>
          </xs:unique>
        </xs:element>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```



```
        <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:complexType>
<xs:complexType name="tBay">
  <xs:complexContent>
    <xs:extension base="tEquipmentContainer">
      <xs:sequence>
        <xs:element name="ConductingEquipment" type="tConductingEquipment" minOccurs="0"
maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInConductingEquipment">
            <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="uniqueChildNameInCE">
            <xs:selector xpath="/scl:SubEquipment|./scl:EqFunction"/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
        <xs:element name="ConnectivityNode" type="tConnectivityNode" minOccurs="0"
maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInConnectivityNode">
            <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:element>
        <xs:element name="Function" type="scl:tFunction" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInFunctionB">
            <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="uniqueChildNameInBayFunc">
            <xs:selector xpath="./*/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

IECNORM.COM Click to view the full PDF of IEC 61850-6:2009/AMD1:2018

```

<xs:complexType name="tLNode">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:attribute name="iedName" type="tIEDNameOrNone" use="optional" default="None"/>
      <xs:attribute name="ldInst" type="tLDInstOrEmpty" use="optional" default=""/>
      <xs:attribute name="prefix" type="tPrefix" use="optional" default=""/>
      <xs:attribute name="lnClass" type="tLNClassEnum" use="required"/>
      <xs:attribute name="lnInst" type="tLNInstOrEmpty" use="optional" default=""/>
      <xs:attribute name="lnType" type="tName" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<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="uniqueLNodeInSubFunction">
            <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="uniqueChildNameInSubFunc">
            <xs:selector xpath="/*"/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
        <xs:element name="GeneralEquipment" type="tGeneralEquipment" minOccurs="0"
maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInGeneralEquipmentOfFunction">
            <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="uniqueChildNameInGEFunc">
            <xs:selector xpath="/*"/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
        <xs:element name="ConductingEquipment" type="tConductingEquipment" minOccurs="0"
maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInConductingEquipmentOfFunction">
            <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="uniqueChildNameInCondEq">
            <xs:selector xpath="/scl:SubEquipment|/scl:EqFunction"/>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
      </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:unique name="uniqueLNodeInGeneralEquipmentOfSubFunction">
            <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="uniqueChildNameInGESubFunc">
  <xs:selector xpath="."/ *>
  <xs:field xpath="@name"/>
</xs:unique>
</xs:element>
<xs:element name="ConductingEquipment" type="scl:tConductingEquipment" minOccurs="0"
maxOccurs="unbounded">
  <xs:unique name="uniqueLNodeInConductingEquipmentOfSubFunction">
    <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="uniqueSubEquipmentSubFunc">
    <xs:selector xpath="./scl:SubEquipment"/>
    <xs:field xpath="@name"/>
  </xs:unique>
</xs:element>
<xs:element name="SubFunction" type="scl:tSubFunction" minOccurs="0" maxOccurs="unbounded">
  <xs:unique name="uniqueLNodeInSubSubFunction">
    <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="uniqueSubSubFunc">
    <xs:selector xpath="./scl:SubFunction"/>
    <xs:field xpath="@name"/>
  </xs:unique>
</xs:element>
</xs:sequence>
<xs:attribute name="type" type="xs:normalizedString" use="optional"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tAbstractEqFuncSubFunc" abstract="true">
  <xs:complexContent>
    <xs:extension base="tPowerSystemResource">
      <xs:sequence>
        <xs:element name="GeneralEquipment" type="tGeneralEquipment" minOccurs="0"
maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInGeneralEquipmentOfFuncForEquipment">
            <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="uniqueChildNameInGEFuncForEquipment">
            <xs:selector xpath="."/ *>
            <xs:field xpath="@name"/>
          </xs:unique>
        </xs:element>
        <xs:element name="EqSubFunction" type="scl:tEqSubFunction" minOccurs="0"
maxOccurs="unbounded">
          <xs:unique name="uniqueLNodeInSubFuncForEquipment">
            <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:element>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

```

        </xs:unique>
        <xs:unique name="uniqueSubFuncForEquipment">
            <xs:selector xpath="./scl:EqSubFunction"/>
            <xs:field xpath="@name"/>
        </xs:unique>
    </xs:element>
</xs:sequence>
<xs:attribute name="type" type="xs:normalizedString" use="optional"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tEqFunction">
    <xs:complexContent>
        <xs:extension base="tAbstractEqFuncSubFunc"/>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="tEqSubFunction">
    <xs:complexContent>
        <xs:extension base="tAbstractEqFuncSubFunc"/>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="tConnectivityNode">
    <xs:complexContent>
        <xs:extension base="tLNodeContainer">
            <xs:attribute name="pathName" type="tConnectivityNodeReference" use="required"/>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="tTerminal">
    <xs:complexContent>
        <xs:extension base="tUnNaming">
            <xs:attribute name="name" type="tAnyName" use="optional" default=""/>
            <xs:attribute name="connectivityNode" type="tConnectivityNodeReference" use="required"/>
            <xs:attribute name="processName" type="tProcessName" use="optional"/>
            <xs:attribute name="substationName" type="tName" use="optional"/>
            <xs:attribute name="voltageLevelName" type="tName" use="optional"/>
            <xs:attribute name="bayName" type="tName" use="optional"/>
            <xs:attribute name="cNodeName" type="tName" use="required"/>
            <xs:attribute name="lineName" type="tName" use="optional"/>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="tGeneralEquipmentContainer" abstract="true">
    <xs:complexContent>
        <xs:extension base="tPowerSystemResource">
            <xs:sequence>
                <xs:element name="GeneralEquipment" type="tGeneralEquipment" minOccurs="0"
maxOccurs="unbounded"/>
                <xs:unique name="uniqueLNodeInGeneralEquipment2">
                    <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="uniqueChildNameInGE2">
                    <xs:selector xpath="./*/>
                    <xs:field xpath="@name"/>
                </xs:unique>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:element name="Function" type="tFunction" minOccurs="0" maxOccurs="unbounded">
    <xs:unique name="uniqueLNodeInFunction">
        <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="uniqueChildNameInFunction">
        <xs:selector xpath="./*/>
        <xs:field xpath="@name"/>
    </xs:unique>
</xs:element>

```

```
        </xs:unique>
      </xs:element>
    </xs:sequence>
  </xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tLine">
  <xs:complexContent>
    <xs:extension base="tGeneralEquipmentContainer">
      <xs:sequence>
        <xs:element name="Voltage" type="tVoltage" minOccurs="0"/>
        <xs:element name="ConductingEquipment" type="scl:tConductingEquipment" maxOccurs="unbounded"/>
        <xs:element name="ConnectivityNode" type="scl:tConnectivityNode" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="type" type="tLineType" use="optional"/>
      <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>
<xs:complexType name="tProcess">
  <xs:complexContent>
    <xs:extension base="tGeneralEquipmentContainer">
      <xs:sequence>
        <xs:element name="ConductingEquipment" type="scl:tConductingEquipment" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="Substation" type="tSubstation" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniqueChildNameInProcessSubstation">
            <xs:selector xpath="./*/>
            <xs:field xpath="@name"/>
          </xs:unique>
          <xs:unique name="uniqueLNodeInProcessSubstation">
            <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:element>
        <xs:element name="Line" type="tLine" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="Process" type="tProcess" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniqueChildNameInSubProcess">
            <xs:selector xpath="./*/>
            <xs:field xpath="@name"/>
          </xs:unique>
          <xs:unique name="uniqueLNodeInSubProcess">
            <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:element>
      </xs:sequence>
      <xs:attribute name="type" type="tProcessType" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

```

<xs:element name="Substation" type="tSubstation">
  <xs:unique name="uniqueChildNameInSubstation">
    <xs:selector xpath="./*/"/>
    <xs:field xpath="@name"/>
  </xs:unique>
  <xs:unique name="uniqueLNodeInSubstation">
    <xs:selector xpath="/scl:LNode"/>
    <xs:field xpath="@InInst"/>
    <xs:field xpath="@InClass"/>
    <xs:field xpath="@iedName"/>
    <xs:field xpath="@IdInst"/>
    <xs:field xpath="@prefix"/>
  </xs:unique>
</xs:element>
<xs:element name="Process" type="tProcess">
  <xs:unique name="uniqueChildNameInProcess">
    <xs:selector xpath="./*/"/>
    <xs:field xpath="@name"/>
  </xs:unique>
  <xs:unique name="uniqueLNodeInProcess">
    <xs:selector xpath="/scl:LNode"/>
    <xs:field xpath="@InInst"/>
    <xs:field xpath="@InClass"/>
    <xs:field xpath="@iedName"/>
    <xs:field xpath="@IdInst"/>
    <xs:field xpath="@prefix"/>
  </xs:unique>
</xs:element>
<xs:element name="Line" type="tLine">
  <xs:unique name="uniqueChildNameInLine">
    <xs:selector xpath="./*/"/>
    <xs:field xpath="@name"/>
  </xs:unique>
  <xs:unique name="uniqueLNodeInLine">
    <xs:selector xpath="/scl:LNode"/>
    <xs:field xpath="@InInst"/>
    <xs:field xpath="@InClass"/>
    <xs:field xpath="@iedName"/>
    <xs:field xpath="@IdInst"/>
    <xs:field xpath="@prefix"/>
  </xs:unique>
</xs:element>
</xs:schema>

```

**<CODE ENDS>**

IECNORM.COM : Click to view the full PDF of IEC 61850-6:2009/AMD1:2018

### A.3 Data type templates

#### File SCL\_DataTypeTemplates.xsd

##### <CODE BEGINS>

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:scl="http://www.iec.ch/61850/2003/SCL" xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns="http://www.iec.ch/61850/2003/SCL" targetNamespace="http://www.iec.ch/61850/2003/SCL"
elementFormDefault="qualified" attributeFormDefault="unqualified" version="2007B4">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      SCL schema version "2007" revision "B" release 4, for IEC 61850-6 Ed. 2.1.

      COPYRIGHT (c) IEC, 2016. All rights reserved. Disclaimer: The IEC disclaims liability for any personal injury,
      property or other damages of any nature whatsoever, whether special, indirect, consequential or compensatory, directly or
      indirectly resulting from this software and the document upon which its methods are based, use of, or reliance upon.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="SCL_BaseTypes.xsd"/>
  <xs:attributeGroup name="agDATrgOp">
    <xs:attribute name="dchg" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="qchg" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="dupd" type="xs:boolean" use="optional" default="false"/>
  </xs:attributeGroup>
  <xs:complexType name="tAbstractDataAttribute" abstract="true">
    <xs:complexContent>
      <xs:extension base="tUnNaming">
        <xs:sequence>
          <xs:element name="Val" type="tVal" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="name" type="tAttributeNameEnum" use="required"/>
        <xs:attribute name="sAddr" type="xs:normalizedString" use="optional"/>
        <xs:attribute name="bType" type="tBasicTypeEnum" use="required"/>
        <xs:attribute name="valKind" type="tValKindEnum" use="optional" default="Set"/>
        <xs:attribute name="type" type="tAnyName" use="optional"/>
        <xs:attribute name="count" type="tDACount" use="optional" default="0"/>
        <xs:attribute name="valImport" type="xs:boolean" use="optional" default="false"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tLNodeType">
    <xs:complexContent>
      <xs:extension base="tIDNaming">
        <xs:sequence>
          <xs:element name="DO" type="tDO" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="iedType" type="tAnyName" use="optional" default=""/>
        <xs:attribute name="lnClass" type="tLNClassEnum" use="required"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tDO">
    <xs:complexContent>
      <xs:extension base="tUnNaming">
        <xs:attribute name="name" type="tDataName" use="required"/>
        <xs:attribute name="type" type="tName" use="required"/>
        <xs:attribute name="accessControl" type="xs:normalizedString" use="optional"/>
        <xs:attribute name="transient" type="xs:boolean" use="optional" default="false"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tDOType">
    <xs:complexContent>
      <xs:extension base="tIDNaming">
        <xs:choice minOccurs="0" maxOccurs="unbounded">
          <xs:element name="SDO" type="tSDO"/>
          <xs:element name="DA" type="tDA">
            <xs:unique name="uniqueProtNsInDA">
              <xs:selector xpath="scl:ProtNs"/>
              <xs:field xpath="@type"/>
              <xs:field xpath="."/>
            </xs:unique>
          </xs:element>
        </xs:choice>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```
</xs:unique>
</xs:element>
</xs:choice>
<xs:attribute name="iedType" type="tAnyName" use="optional" default=""/>
<xs:attribute name="cdc" type="tCDCEnum" use="required"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tSDO">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:attribute name="name" type="tSubDataName" use="required"/>
      <xs:attribute name="type" type="tName" use="required"/>
      <xs:attribute name="count" type="tSDOCount" use="optional" default="0"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tDA">
  <xs:complexContent>
    <xs:extension base="tAbstractDataAttribute">
      <xs:sequence>
        <xs:element name="ProtNs" type="tProtNs" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attributeGroup ref="agDATrgOp"/>
      <xs:attribute name="fc" type="tFCEnum" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tDAType">
  <xs:complexContent>
    <xs:extension base="tIDNaming">
      <xs:sequence>
        <xs:element name="BDA" type="tBDA" maxOccurs="unbounded"/>
        <xs:element name="ProtNs" type="tProtNs" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="iedType" type="tAnyName" use="optional" default=""/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tBDA">
  <xs:complexContent>
    <xs:extension base="tAbstractDataAttribute"/>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tEnumType">
  <xs:complexContent>
    <xs:extension base="tIDNaming">
      <xs:sequence>
        <xs:element name="EnumVal" type="tEnumVal" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tProtNs">
  <xs:simpleContent>
    <xs:extension base="tNamespaceName">
      <xs:attribute name="type" use="optional" default="8-MMS">
        <xs:simpleType>
          <xs:restriction base="xs:normalizedString">
            <xs:minLength value="1"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tEnumVal">
  <xs:simpleContent>
    <xs:extension base="tEnumStringValue">
      <xs:attribute name="ord" type="xs:int" use="required"/>
      <xs:attributeGroup ref="agDesc"/>
    </xs:extension>
  </xs:simpleContent>
```



```

</xs:complexType>
<xs:complexType name="tDataTypeTemplates">
  <xs:sequence>
    <xs:element name="LNodeType" type="tLNodeType" maxOccurs="unbounded">
      <xs:unique name="uniqueDOIInLNodeType">
        <xs:selector xpath="scl:DO"/>
        <xs:field xpath="@name"/>
      </xs:unique>
    </xs:element>
    <xs:element name="DOType" type="tDOType" maxOccurs="unbounded">
      <xs:unique name="uniqueDAorSDOInDOType">
        <xs:selector xpath="/*"/>
        <xs:field xpath="@name"/>
      </xs:unique>
    </xs:element>
    <xs:element name="DAType" type="tDAType" minOccurs="0" maxOccurs="unbounded">
      <xs:unique name="uniqueBDAInDAType">
        <xs:selector xpath="scl:BDA"/>
        <xs:field xpath="@name"/>
      </xs:unique>
      <xs:unique name="uniqueProtNs">
        <xs:selector xpath="scl:ProtNs"/>
        <xs:field xpath="@type"/>
        <xs:field xpath="."/>
      </xs:unique>
    </xs:element>
    <xs:element name="EnumType" type="tEnumType" minOccurs="0" maxOccurs="unbounded">
      <xs:unique name="uniqueOrdInEnumType">
        <xs:selector xpath="scl:EnumVal"/>
        <xs:field xpath="@ord"/>
      </xs:unique>
      <xs:unique name="uniqueEnumValue">
        <xs:selector xpath="scl:EnumVal"/>
        <xs:field xpath="."/>
      </xs:unique>
    </xs:element>
  </xs:sequence>
</xs:complexType>
<xs:element name="DataTypeTemplates" type="tDataTypeTemplates">
  <xs:key name="DOTypeKey">
    <xs:selector xpath="scl:DOType"/>
    <xs:field xpath="@id"/>
  </xs:key>
  <xs:keyref name="ref2DOType" refer="DOTypeKey">
    <xs:selector xpath="scl:LNodeType/scl:DO"/>
    <xs:field xpath="@type"/>
  </xs:keyref>
  <xs:keyref name="ref2DOTypeForSDO" refer="DOTypeKey">
    <xs:selector xpath="scl:DOType/scl:SDO"/>
    <xs:field xpath="@type"/>
  </xs:keyref>
  <xs:unique name="uniqueDTT_ID">
    <xs:selector xpath="*/>
    <xs:field xpath="@id"/>
  </xs:unique>
  <xs:key name="EnumTypeDATypeKey">
    <xs:selector xpath="scl:DAType | scl:EnumType"/>
    <xs:field xpath="@id"/>
  </xs:key>
  <xs:keyref name="ref2EnumTypeDAType" refer="scl:EnumTypeDATypeKey">
    <xs:selector xpath="scl:DOType/scl:DA | scl:DAType/scl:BDA"/>
    <xs:field xpath="@type"/>
  </xs:keyref>
</xs:element>
</xs:schema>

```

**<CODE ENDS>**

## A.4 IED capabilities and structure

### File SCL\_IED.xsd

#### <CODE BEGINS>

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:scl="http://www.iec.ch/61850/2003/SCL" xmlns="http://www.iec.ch/61850/2003/SCL"
xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="http://www.iec.ch/61850/2003/SCL"
elementFormDefault="qualified" attributeFormDefault="unqualified" version="2007B4">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      SCL schema version "2007" revision "B" release 4, for IEC 61850-6 Ed. 2.1.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="SCL_BaseTypes.xsd"/>
  <xs:attributeGroup name="agAuthentication">
    <xs:attribute name="none" type="xs:boolean" use="optional" default="true"/>
    <xs:attribute name="password" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="weak" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="strong" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="certificate" type="xs:boolean" use="optional" default="false"/>
  </xs:attributeGroup>
  <xs:attributeGroup name="agSmvOpts">
    <xs:attribute name="refreshTime" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="sampleSynchronized" type="xs:boolean" use="optional" fixed="true"/>
    <xs:attribute name="sampleRate" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="dataSet" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="security" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="timestamp" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="synchSourceId" type="xs:boolean" use="optional" default="false"/>
  </xs:attributeGroup>
  <xs:attributeGroup name="agOptFields">
    <xs:attribute name="seqNum" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="timeStamp" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="dataSet" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="reasonCode" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="dataRef" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="entryID" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="configRef" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="bufOvfl" type="xs:boolean" use="optional" default="true"/>
  </xs:attributeGroup>
  <xs:attributeGroup name="agLDRef">
    <xs:attributeGroup ref="scl:agDesc"/>
    <xs:attribute name="iedName" type="tIEDName" use="required"/>
    <xs:attribute name="ldInst" type="tLDInst" use="required"/>
  </xs:attributeGroup>
  <xs:attributeGroup name="agLNRef">
    <xs:attributeGroup ref="agLDRef"/>
    <xs:attribute name="prefix" type="tPrefix" use="optional" default=""/>
    <xs:attribute name="lnClass" type="tLNClassEnum" use="required"/>
    <xs:attribute name="lnInst" type="tLNInstOrEmpty" use="required"/>
  </xs:attributeGroup>
  <xs:complexType name="tIED">
    <xs:complexContent>
      <xs:extension base="tUnNaming">
        <xs:sequence>
          <xs:element name="Services" type="tServices" minOccurs="0"/>
          <xs:element name="AccessPoint" type="tAccessPoint" maxOccurs="unbounded">
            <xs:unique name="uniqueLNInAccessPoint">
              <xs:selector xpath="/scl:LN"/>
              <xs:field xpath="@inst"/>
              <xs:field xpath="@lnClass"/>
              <xs:field xpath="@prefix"/>
            </xs:unique>
          </xs:element>
          <xs:element name="KDC" type="tKDC" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```
</xs:sequence>
<xs:attribute name="name" type="tIEDName" use="required"/>
<xs:attribute name="type" type="xs:normalizedString" use="optional"/>
<xs:attribute name="manufacturer" type="xs:normalizedString" use="optional"/>
<xs:attribute name="configVersion" type="xs:normalizedString" use="optional"/>
<xs:attribute name="originalSclVersion" type="tSclVersion" use="optional" default="2003"/>
<xs:attribute name="originalSclRevision" type="tSclRevision" use="optional" default="A"/>
<xs:attribute name="originalSclRelease" type="tSclRelease" use="optional" default="1"/>
<xs:attribute name="engRight" type="tRightEnum" use="optional" default="full"/>
<xs:attribute name="owner" type="xs:normalizedString" use="optional"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tServices">
  <xs:all>
    <xs:element name="DynAssociation" type="scl:tServiceWithOptionalMax" minOccurs="0"/>
    <xs:element name="SettingGroups" type="scl:tSettingGroups" minOccurs="0"/>
    <xs:element name="GetDirectory" type="scl:tServiceYesNo" minOccurs="0"/>
    <xs:element name="GetDataObjectDefinition" type="scl:tServiceYesNo" minOccurs="0"/>
    <xs:element name="DataObjectDirectory" type="scl:tServiceYesNo" minOccurs="0"/>
    <xs:element name="GetDataSetValue" type="scl:tServiceYesNo" minOccurs="0"/>
    <xs:element name="DataSetValue" type="scl:tServiceYesNo" minOccurs="0"/>
    <xs:element name="DataSetDirectory" type="scl:tServiceYesNo" minOccurs="0"/>
    <xs:element name="ConfDataSet" type="scl:tServiceForConfDataSet" minOccurs="0"/>
    <xs:element name="DynDataSet" type="scl:tServiceWithMaxAndMaxAttributes" minOccurs="0"/>
    <xs:element name="ReadWrite" type="scl:tServiceYesNo" minOccurs="0"/>
    <xs:element name="TimerActivatedControl" type="scl:tServiceYesNo" minOccurs="0"/>
    <xs:element name="ConfReportControl" type="scl:tServiceConfReportControl" minOccurs="0"/>
    <xs:element name="GetCBValues" type="scl:tServiceYesNo" minOccurs="0"/>
    <xs:element name="ConfLogControl" type="scl:tServiceWithMaxNonZero" minOccurs="0"/>
    <xs:element name="ReportSettings" type="scl:tReportSettings" minOccurs="0"/>
    <xs:element name="LogSettings" type="scl:tLogSettings" minOccurs="0"/>
    <xs:element name="GSESettings" type="scl:tGSESettings" minOccurs="0"/>
    <xs:element name="SMVSettings" type="scl:tSMVSettings" minOccurs="0"/>
    <xs:element name="GSEDir" type="scl:tServiceYesNo" minOccurs="0"/>
    <xs:element name="GOOSE" type="scl:tGOOSECapabilities" minOccurs="0"/>
    <xs:element name="GSSE" type="scl:tServiceWithMax" minOccurs="0"/>
    <xs:element name="SMVsc" type="scl:tSMVsc" minOccurs="0"/>
    <xs:element name="FileHandling" type="scl:tFileHandling" minOccurs="0"/>
    <xs:element name="ConfLNs" type="scl:tConfLNs" minOccurs="0"/>
    <xs:element name="ClientServices" type="scl:tClientServices" minOccurs="0"/>
    <xs:element name="ConfLdName" type="scl:tServiceYesNo" minOccurs="0"/>
    <xs:element name="SupSubscription" type="scl:tSupSubscription" minOccurs="0"/>
    <xs:element name="ConfSigRer" type="scl:tServiceWithMaxNonZero" minOccurs="0"/>
    <xs:element name="ValueHandling" type="scl:tValueHandling" minOccurs="0"/>
    <xs:element name="RedProt" type="scl:tRedProt" minOccurs="0"/>
    <xs:element name="TimeSyncProt" type="scl:tTimeSyncProt" minOccurs="0"/>
    <xs:element name="CommProt" type="scl:tCommProt" minOccurs="0"/>
  </xs:all>
  <xs:attribute name="nameLength" use="optional" default="32">
    <xs:simpleType>
      <xs:restriction base="xs:token">
        <xs:pattern value="32"/>
        <xs:pattern value="64"/>
        <xs:pattern value="6[5-9]"/>
        <xs:pattern value="[7-9]d"/>
        <xs:pattern value="[1-9]d\d+"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:complexType>
<xs:complexType name="tAccessPoint">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <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:choice>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

```

    <xs:element name="ServerAt" type="tServerAt"/>
  </xs:choice>
  <xs:element name="Services" type="scl: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="name" type="tAccessPointName" use="required"/>
<xs:attribute name="router" type="xs:boolean" use="optional" default="false"/>
<xs:attribute name="clock" type="xs:boolean" use="optional" default="false"/>
<xs:attribute name="kdc" type="xs:boolean" use="optional" default="false"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tCertificate">
  <xs:complexContent>
    <xs:extension base="tNaming">
      <xs:sequence>
        <xs:element name="Subject" type="tCert"/>
        <xs:element name="IssuerName" type="tCert"/>
      </xs:sequence>
      <xs:attribute name="xferNumber" type="xs:unsignedInt" use="optional"/>
      <xs:attribute name="serialNumber" use="required">
        <xs:simpleType>
          <xs:restriction base="xs:normalizedString">
            <xs:minLength value="1"/>
            <xs:pattern value="[0-9]+"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tCert">
  <xs:attribute name="commonName" use="required">
    <xs:simpleType>
      <xs:restriction base="xs:normalizedString">
        <xs:minLength value="4"/>
        <xs:pattern value="none"/>
        <xs:pattern value="CN=.\+"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="idHierarchy" use="required">
    <xs:simpleType>
      <xs:restriction base="xs:normalizedString">
        <xs:minLength value="1"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:complexType>
<xs:complexType name="tServerAt">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:attribute name="apName" type="tAccessPointName" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tServer">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:sequence>
        <xs:element name="Authentication">
          <xs:complexType>
            <xs:attributeGroup ref="agAuthentication"/>
          </xs:complexType>
        </xs:element>
        <xs:element name="LDevice" type="tLDevice" maxOccurs="unbounded">
          <xs:unique name="uniqueLNInLDevice">
            <xs:selector xpath="/scl:LN"/>
            <xs:field xpath="@inst"/>
            <xs:field xpath="@lnClass"/>
            <xs:field xpath="@prefix"/>
          </xs:unique>
        </xs:element>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```



```
        </xs:element>
        <xs:element name="Association" type="tAssociation" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="timeout" type="xs:unsignedInt" use="optional" default="30"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tLDDevice">
    <xs:complexContent>
        <xs:extension base="tUnNaming">
            <xs:sequence>
                <xs:element ref="LN0"/>
                <xs:element ref="LN" minOccurs="0" maxOccurs="unbounded"/>
                <xs:element name="AccessControl" type="tAccessControl" minOccurs="0"/>
            </xs:sequence>
            <xs:attribute name="inst" type="tLDInst" use="required"/>
            <xs:attribute name="ldName" type="tLDName" use="optional"/>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="tAccessControl" mixed="true">
    <xs:complexContent>
        <xs:extension base="tAnyContentFromOtherNamespace"/>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="tAssociation">
    <xs:attributeGroup ref="agLNRef"/>
    <xs:attribute name="kind" type="tAssociationKindEnum" use="required"/>
    <xs:attribute name="associationID" type="tAssociationID" use="optional"/>
</xs:complexType>
<xs:element name="LN0">
    <xs:complexType>
        <xs:complexContent>
            <xs:extension base="tLN0"/>
        </xs:complexContent>
    </xs:complexType>
    <xs:unique name="uniqueReportControlInLN0">
        <xs:selector xpath="/scl:ReportControl"/>
        <xs:field xpath="@name"/>
    </xs:unique>
    <xs:unique name="uniqueLogControlInLN0">
        <xs:selector xpath="/scl:LogControl"/>
        <xs:field xpath="@name"/>
    </xs:unique>
    <xs:unique name="uniqueGSEControlInLN0">
        <xs:selector xpath="/scl:GSEControl"/>
        <xs:field xpath="@name"/>
    </xs:unique>
    <xs:unique name="uniqueSampledValueControlInLN0">
        <xs:selector xpath="/scl:SampledValueControl"/>
        <xs:field xpath="@name"/>
    </xs:unique>
    <xs:key name="DataSetKeyLN0">
        <xs:selector xpath="/scl:DataSet"/>
        <xs:field xpath="@name"/>
    </xs:key>
    <xs:keyref name="ref2DataSetReportLN0" refer="DataSetKeyLN0">
        <xs:selector xpath="/scl:ReportControl"/>
        <xs:field xpath="@datSet"/>
    </xs:keyref>
    <xs:keyref name="ref2DataSetLogLN0" refer="DataSetKeyLN0">
        <xs:selector xpath="/scl:LogControl"/>
        <xs:field xpath="@datSet"/>
    </xs:keyref>
    <xs:keyref name="ref2DataSetGSELN0" refer="DataSetKeyLN0">
        <xs:selector xpath="/scl:GSEControl"/>
        <xs:field xpath="@datSet"/>
    </xs:keyref>
    <xs:keyref name="ref2DataSetSVLN0" refer="DataSetKeyLN0">
        <xs:selector xpath="/scl:SampledValueControl"/>
        <xs:field xpath="@datSet"/>
    </xs:keyref>
    <xs:unique name="uniqueDOIinLN0">
```



```

    <xs:selector xpath="/scl:DOI"/>
    <xs:field xpath="@name"/>
  </xs:unique>
  <xs:unique name="uniqueLogInLN0">
    <xs:selector xpath="/scl:Log"/>
    <xs:field xpath="@name"/>
  </xs:unique>
</xs:element>
<xs:element name="LN" type="tLN">
  <xs:unique name="uniqueReportControlInLN">
    <xs:selector xpath="/scl:ReportControl"/>
    <xs:field xpath="@name"/>
  </xs:unique>
  <xs:unique name="uniqueLogControlInLN">
    <xs:selector xpath="/scl:LogControl"/>
    <xs:field xpath="@name"/>
  </xs:unique>
  <xs:key name="DataSetKeyInLN">
    <xs:selector xpath="/scl:DataSet"/>
    <xs:field xpath="@name"/>
  </xs:key>
  <xs:keyref name="ref2DataSetReport" refer="DataSetKeyInLN">
    <xs:selector xpath="/scl:ReportControl"/>
    <xs:field xpath="@datSet"/>
  </xs:keyref>
  <xs:keyref name="ref2DataSetLog" refer="DataSetKeyInLN">
    <xs:selector xpath="/scl:LogControl"/>
    <xs:field xpath="@datSet"/>
  </xs:keyref>
  <xs:unique name="uniqueDOIinLN">
    <xs:selector xpath="/scl:DOI"/>
    <xs:field xpath="@name"/>
  </xs:unique>
  <xs:unique name="uniqueLogInLN">
    <xs:selector xpath="/scl:Log"/>
    <xs:field xpath="@name"/>
  </xs:unique>
</xs:element>
<xs:complexType name="tAnyLN" abstract="true">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:sequence>
        <xs:element name="DataSet" type="tDataSet" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="ReportControl" type="tReportControl" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="LogControl" type="tLogControl" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="DOI" type="tDOI" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniqueSDI_DAIinDOI">
            <xs:selector xpath="/scl:DAI|/scl:SDI"/>
            <xs:field xpath="@name"/>
            <xs:field xpath="@ix"/>
          </xs:unique>
        </xs:element>
        <xs:element name="Inputs" type="tInputs" minOccurs="0">
          <!--<xs:unique name="uniqueExtRefInInputs">
            <xs:selector xpath="/scl:ExtRef"/>
            <xs:field xpath="@iedName"/>
            <xs:field xpath="@ldInst"/>
            <xs:field xpath="@prefix"/>
            <xs:field xpath="@lnClass"/>
            <xs:field xpath="@lnInst"/>
            <xs:field xpath="@doName"/>
            <xs:field xpath="@daName"/>
            <xs:field xpath="@intAddr"/>
          </xs:unique-->
        </xs:element>
        <xs:element name="Log" type="scl:tLog" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="lnType" type="tName" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tLN">
  <xs:complexContent>

```



```

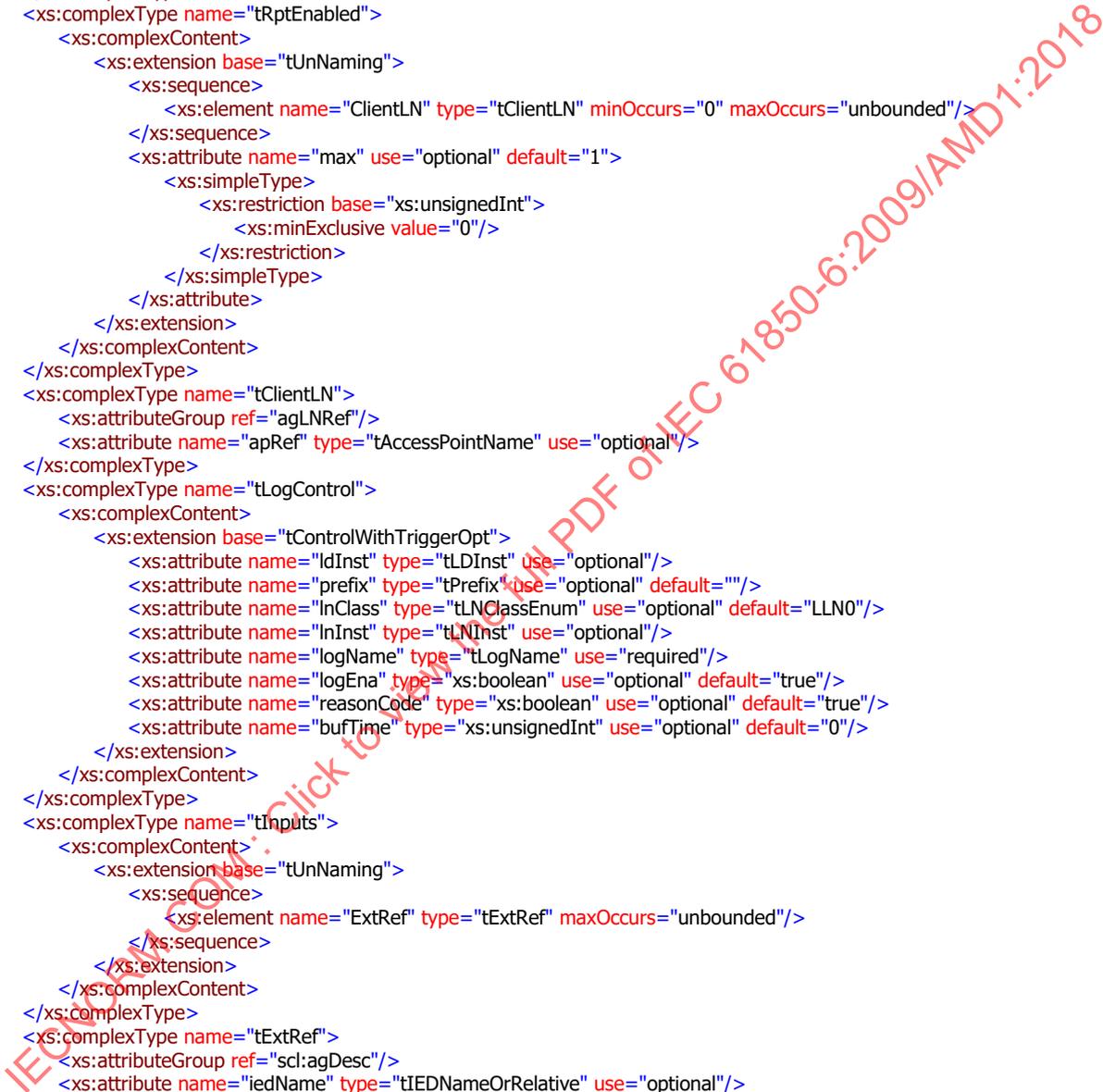
    <xs:extension base="tAnyLN">
      <xs:attribute name="prefix" type="tPrefix" use="optional" default=""/>
      <xs:attribute name="InClass" type="tLNClassEnum" use="required"/>
      <xs:attribute name="inst" type="tLNInst" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tLN0">
  <xs:complexContent>
    <xs:extension base="tAnyLN">
      <xs:sequence>
        <xs:element name="GSEControl" type="tGSEControl" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="SampledValueControl" type="tSampledValueControl" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="SettingControl" type="tSettingControl" minOccurs="0"/>
      </xs:sequence>
      <xs:attribute name="InClass" type="tLNClassEnum" use="required" fixed="LLN0"/>
      <xs:attribute name="inst" type="xs:normalizedString" use="required" fixed=""/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tDataSet">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:choice maxOccurs="unbounded">
        <xs:element name="FCDA" type="tFCDA"/>
      </xs:choice>
      <xs:attribute name="name" type="tDataSetName" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tFCDA">
  <xs:attribute name="IdInst" type="tLDInst" use="optional"/>
  <xs:attribute name="prefix" type="tPrefix" use="optional" default=""/>
  <xs:attribute name="InClass" type="tLNClassEnum" use="optional"/>
  <xs:attribute name="InInst" type="tLNInst" use="optional"/>
  <xs:attribute name="doName" type="tFullDOName" use="optional"/>
  <xs:attribute name="daName" type="tFullAttributeName" use="optional"/>
  <xs:attribute name="fc" type="tFCEnum" use="required"/>
  <xs:attribute name="ix" type="xs:unsignedInt" use="optional"/>
</xs:complexType>
<xs:complexType name="tControl" abstract="true">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:attribute name="name" type="tCBName" use="required"/>
      <xs:attribute name="datSet" type="tDataSetName" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tControlWithTriggerOpt" abstract="true">
  <xs:complexContent>
    <xs:extension base="tControl">
      <xs:sequence>
        <xs:element name="TrgOps" type="tTrgOps" minOccurs="0"/>
      </xs:sequence>
      <xs:attribute name="intgPd" type="xs:unsignedInt" use="optional" default="0"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tTrgOps">
  <xs:attribute name="dchg" type="xs:boolean" use="optional" default="false"/>
  <xs:attribute name="qchg" type="xs:boolean" use="optional" default="false"/>
  <xs:attribute name="dupd" type="xs:boolean" use="optional" default="false"/>
  <xs:attribute name="period" type="xs:boolean" use="optional" default="false"/>
  <xs:attribute name="gi" type="xs:boolean" use="optional" default="true"/>
</xs:complexType>
<xs:complexType name="tReportControl">
  <xs:complexContent>
    <xs:extension base="tControlWithTriggerOpt">
      <xs:sequence>
        <xs:element name="OptFields">
          <xs:complexType>
            <xs:attributeGroup ref="agOptFields"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

```

    </xs:complexType>
  </xs:element>
  <xs:element name="RptEnabled" type="tRptEnabled" minOccurs="0"/>
</xs:sequence>
<xs:attribute name="rptID" type="tMessageID" use="optional"/>
<xs:attribute name="confRev" type="xs:unsignedInt" use="required"/>
<xs:attribute name="buffered" type="xs:boolean" use="optional" default="false"/>
<xs:attribute name="bufTime" type="xs:unsignedInt" use="optional" default="0"/>
<xs:attribute name="indexed" type="xs:boolean" use="optional" default="true"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tRptEnabled">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:sequence>
        <xs:element name="ClientLN" type="tClientLN" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="max" use="optional" default="1">
        <xs:simpleType>
          <xs:restriction base="xs:unsignedInt">
            <xs:minExclusive value="0"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tClientLN">
  <xs:attributeGroup ref="agLNRef"/>
  <xs:attribute name="apRef" type="tAccessPointName" use="optional"/>
</xs:complexType>
<xs:complexType name="tLogControl">
  <xs:complexContent>
    <xs:extension base="tControlWithTriggerOpt">
      <xs:attribute name="ldInst" type="tLDInst" use="optional"/>
      <xs:attribute name="prefix" type="tPrefix" use="optional" default=""/>
      <xs:attribute name="lnClass" type="tLNClassEnum" use="optional" default="LLN0"/>
      <xs:attribute name="lnInst" type="tLNInst" use="optional"/>
      <xs:attribute name="logName" type="tLogName" use="required"/>
      <xs:attribute name="logEna" type="xs:boolean" use="optional" default="true"/>
      <xs:attribute name="reasonCode" type="xs:boolean" use="optional" default="true"/>
      <xs:attribute name="bufTime" type="xs:unsignedInt" use="optional" default="0"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tInputs">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:sequence>
        <xs:element name="ExtRef" type="tExtRef" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tExtRef">
  <xs:attributeGroup ref="scl:agDesc"/>
  <xs:attribute name="iedName" type="tIEDNameOrRelative" use="optional"/>
  <xs:attribute name="ldInst" type="tLDInst" use="optional"/>
  <xs:attribute name="prefix" type="tPrefix" use="optional"/>
  <xs:attribute name="lnClass" type="tLNClassEnum" use="optional"/>
  <xs:attribute name="lnInst" type="tLNInst" use="optional"/>
  <xs:attribute name="doName" type="tFullDOName" use="optional"/>
  <xs:attribute name="daName" type="tFullAttributeName" use="optional"/>
  <xs:attribute name="intAddr" type="xs:normalizedString" use="optional"/>
  <xs:attribute name="serviceType" type="tServiceType" use="optional"/>
  <xs:attribute name="srcLDInst" type="tLDInst" use="optional"/>
  <xs:attribute name="srcPrefix" type="tPrefix" use="optional"/>
  <xs:attribute name="srcLNClass" type="tLNClassEnum" use="optional"/>
  <xs:attribute name="srcLNInst" type="tLNInst" use="optional"/>
  <xs:attribute name="srcCBName" type="tCBName" use="optional"/>
  <xs:attribute name="pServT" type="tServiceType" use="optional"/>
  <xs:attribute name="pLN" type="tLNClassEnum" use="optional"/>

```



```

    <xs:attribute name="pDO" type="tFullIDName" use="optional"/>
    <xs:attribute name="pDA" type="tFullAttributeName" use="optional"/>
  </xs:complexType>
  <xs:complexType name="tLog">
    <xs:complexContent>
      <xs:extension base="tUnNaming">
        <xs:attribute name="name" type="tLogName" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tControlWithIEDName">
    <xs:complexContent>
      <xs:extension base="tControl">
        <xs:sequence>
          <xs:element name="IEDName" minOccurs="0" maxOccurs="unbounded">
            <xs:complexType>
              <xs:simpleContent>
                <xs:extension base="tIEDName">
                  <xs:attribute name="apRef" type="tAccessPointName" use="optional"/>
                  <xs:attribute name="ldInst" type="tLDInst" use="optional"/>
                  <xs:attribute name="prefix" type="tPrefix" use="optional"/>
                  <xs:attribute name="lnClass" type="tLNClassEnum" use="optional"/>
                  <xs:attribute name="lnInst" type="tLNInst" use="optional"/>
                </xs:extension>
              </xs:simpleContent>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
        <xs:attribute name="confRev" type="xs:unsignedInt" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tProtocol">
    <xs:simpleContent>
      <xs:extension base="xs:normalizedString">
        <xs:attribute name="mustUnderstand" type="xs:boolean" use="required" fixed="true"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="tGSEControl">
    <xs:complexContent>
      <xs:extension base="tControlWithIEDName">
        <xs:sequence>
          <xs:element name="Protocol" type="tProtocol" fixed="R-GOOSE" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="type" type="tGSEControlTypeEnum" use="optional" default="GOOSE"/>
        <xs:attribute name="appID" type="tMessageID" use="required"/>
        <xs:attribute name="fixedOffs" type="xs:boolean" use="optional" default="false"/>
        <xs:attribute name="securityEnable" type="scl:tPredefinedTypeOfSecurityEnum" use="optional"
default="None"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tSampledValueControl">
    <xs:complexContent>
      <xs:extension base="tControlWithIEDName">
        <xs:sequence>
          <xs:element name="SmvOpts">
            <xs:complexType>
              <xs:attributeGroup ref="agSmvOpts"/>
            </xs:complexType>
          </xs:element>
          <xs:element name="Protocol" type="tProtocol" fixed="R-SV" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="smvID" type="tMessageID" use="required"/>
        <xs:attribute name="multicast" type="xs:boolean" default="true"/>
        <xs:attribute name="smpRate" type="xs:unsignedInt" use="required"/>
        <xs:attribute name="nofASDU" type="xs:unsignedInt" use="required"/>
        <xs:attribute name="smpMod" type="tSmpMod" use="optional" default="SmpPerPeriod"/>
        <xs:attribute name="securityEnable" type="tPredefinedTypeOfSecurityEnum" use="optional"
default="None"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

</xs:complexType>
<xs:complexType name="tSettingControl">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:attribute name="numOfSGs" use="required">
        <xs:simpleType>
          <xs:restriction base="xs:unsignedInt">
            <xs:minInclusive value="1"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
      <xs:attribute name="actSG" use="optional" default="1">
        <xs:simpleType>
          <xs:restriction base="xs:unsignedInt">
            <xs:minInclusive value="1"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
      <xs:attribute name="resvTms" type="xs:unsignedShort" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tDOI">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element name="SDI" type="tSDI">
          <xs:unique name="uniqueSDI_DAIinSDI">
            <xs:selector xpath="./scl:DAI./scl:SDI"/>
            <xs:field xpath="@name"/>
            <xs:field xpath="@ix"/>
          </xs:unique>
        </xs:element>
        <xs:element name="DAI" type="tDAI"/>
      </xs:choice>
      <xs:attribute name="name" type="tDataName" use="required"/>
      <xs:attribute name="ix" type="xs:unsignedInt" use="optional"/>
      <xs:attribute name="accessControl" type="xs:normalizedString" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tSDI">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element name="SDI" type="tSDI"/>
        <xs:element name="DAI" type="tDAI"/>
      </xs:choice>
      <xs:attribute name="name" type="tAttributeNameEnum" use="required"/>
      <xs:attribute name="ix" type="xs:unsignedInt" use="optional"/>
      <xs:attribute name="sAddr" type="xs:normalizedString" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tDAI">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:sequence>
        <xs:element name="Val" type="tVal" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="name" type="tAttributeNameEnum" use="required"/>
      <xs:attribute name="sAddr" type="xs:normalizedString" use="optional"/>
      <xs:attribute name="valKind" type="tValKindEnum" use="optional"/>
      <xs:attribute name="ix" type="xs:unsignedInt" use="optional"/>
      <xs:attribute name="valImport" type="xs:boolean" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tServiceYesNo"/>
<xs:complexType name="tServiceWithOptionalMax">
  <xs:attribute name="max" type="xs:unsignedInt" use="optional"/>
</xs:complexType>
<xs:complexType name="tServiceWithMax">

```



```

    <xs:attribute name="max" type="xs:unsignedInt" use="required"/>
  </xs:complexType>
  <xs:complexType name="tServiceWithMaxNonZero">
    <xs:attribute name="max" use="required">
      <xs:simpleType>
        <xs:restriction base="xs:unsignedInt">
          <xs:minExclusive value="0"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:complexType>
  <xs:complexType name="tServiceConfReportControl">
    <xs:complexContent>
      <xs:extension base="tServiceWithMax">
        <xs:attribute name="bufMode" use="optional" default="both">
          <xs:simpleType>
            <xs:restriction base="xs:Name">
              <xs:enumeration value="unbuffered"/>
              <xs:enumeration value="buffered"/>
              <xs:enumeration value="both"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="bufConf" type="xs:boolean" use="optional" default="false"/>
        <xs:attribute name="maxBuf" type="xs:unsignedInt" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tServiceWithMaxAndMaxAttributes">
    <xs:complexContent>
      <xs:extension base="tServiceWithMax">
        <xs:attribute name="maxAttributes" use="optional">
          <xs:simpleType>
            <xs:restriction base="xs:unsignedInt">
              <xs:minExclusive value="0"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tServiceWithMaxAndModify">
    <xs:complexContent>
      <xs:extension base="tServiceWithMax">
        <xs:attribute name="modify" type="xs:boolean" use="optional" default="true"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tServiceForConfDataSet">
    <xs:complexContent>
      <xs:extension base="tServiceWithMaxAndMaxAttributes">
        <xs:attribute name="modify" type="xs:boolean" use="optional" default="true"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tClientServices">
    <xs:sequence>
      <xs:element name="TimeSyncProt" type="scl:tTimeSyncProt" minOccurs="0"/>
      <xs:element name="McSecurity" type="scl:tMcSecurity" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="goose" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="gsse" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="bufReport" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="unbufReport" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="readLog" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="sv" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="supportsLdName" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="maxAttributes" use="optional">
      <xs:simpleType>
        <xs:restriction base="xs:unsignedInt"/>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="maxReports" use="optional">

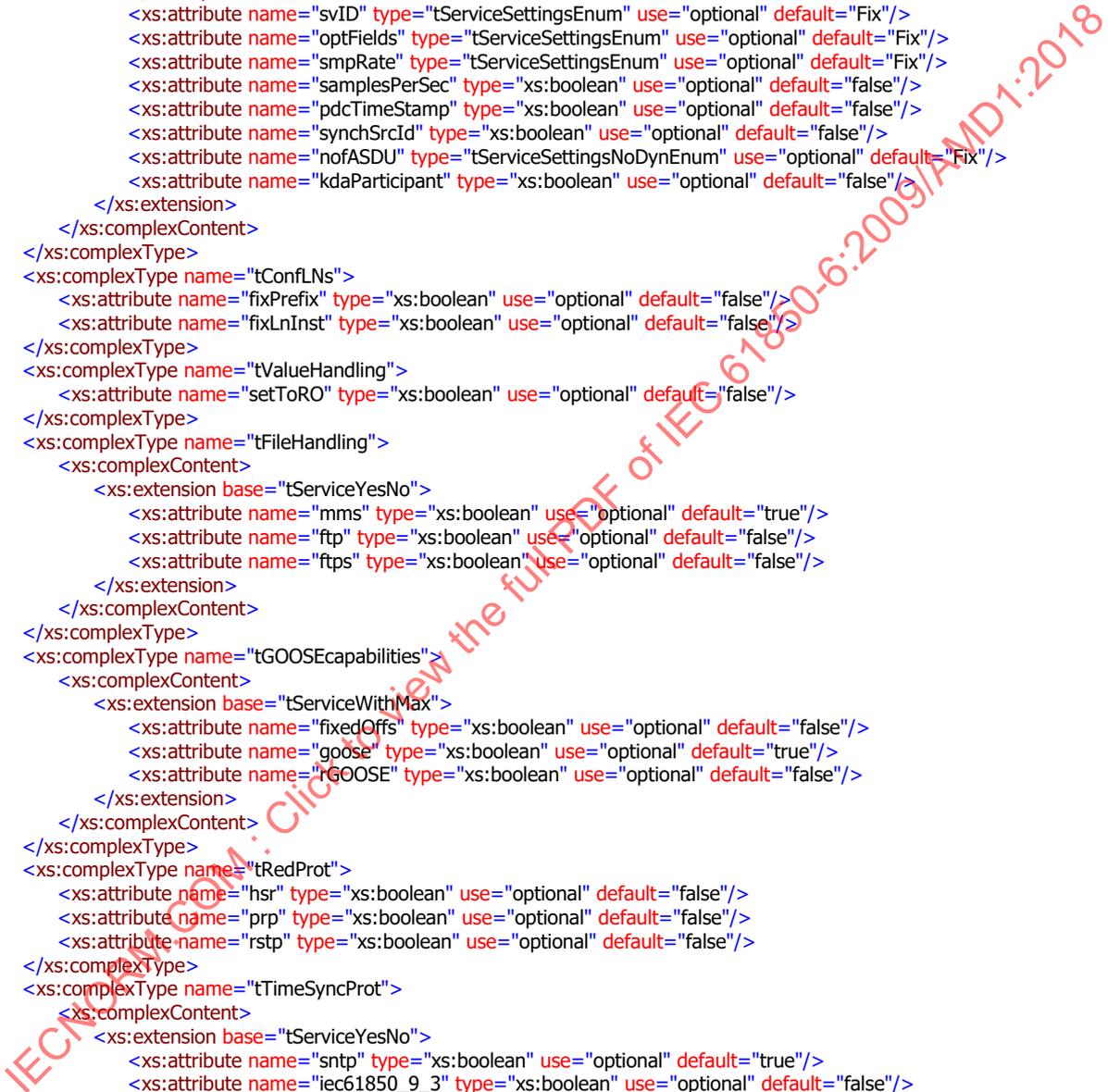
```

```

    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt"/>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="maxGOOSE" use="optional">
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt"/>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="maxSMV" use="optional">
    <xs:simpleType>
      <xs:restriction base="xs:unsignedInt"/>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="rGOOSE" type="xs:boolean" use="optional" default="false"/>
  <xs:attribute name="rSV" type="xs:boolean" use="optional" default="false"/>
  <xs:attribute name="noIctBinding" type="xs:boolean" use="optional" default="false"/>
</xs:complexType>
<xs:complexType name="tServiceSettings" abstract="true">
  <xs:attribute name="cbName" type="tServiceSettingsNoDynEnum" use="optional" default="Fix"/>
  <xs:attribute name="datSet" type="tServiceSettingsEnum" use="optional" default="Fix"/>
</xs:complexType>
<xs:complexType name="tReportSettings">
  <xs:complexContent>
    <xs:extension base="tServiceSettings">
      <xs:attribute name="rptID" type="tServiceSettingsEnum" use="optional" default="Fix"/>
      <xs:attribute name="optFields" type="tServiceSettingsEnum" use="optional" default="Fix"/>
      <xs:attribute name="bufTime" type="tServiceSettingsEnum" use="optional" default="Fix"/>
      <xs:attribute name="trgOps" type="tServiceSettingsEnum" use="optional" default="Fix"/>
      <xs:attribute name="intgPd" type="tServiceSettingsEnum" use="optional" default="Fix"/>
      <xs:attribute name="resvTms" type="xs:boolean" use="optional" default="false"/>
      <xs:attribute name="owner" type="xs:boolean" use="optional" default="false"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tLogSettings">
  <xs:complexContent>
    <xs:extension base="tServiceSettings">
      <xs:attribute name="logEna" type="tServiceSettingsEnum" use="optional" default="Fix"/>
      <xs:attribute name="trgOps" type="tServiceSettingsEnum" use="optional" default="Fix"/>
      <xs:attribute name="intgPd" type="tServiceSettingsEnum" use="optional" default="Fix"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tGSESettings">
  <xs:complexContent>
    <xs:extension base="tServiceSettings">
      <xs:sequence>
        <xs:element name="McSecurity" type="scl:tMcSecurity" minOccurs="0"/>
      </xs:sequence>
      <xs:attribute name="appID" type="tServiceSettingsEnum" use="optional" default="Fix"/>
      <xs:attribute name="dataLabel" type="tServiceSettingsEnum" use="optional" default="Fix"/>
      <xs:attribute name="kdaParticipant" type="xs:boolean" use="optional" default="false"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tSMVSettings">
  <xs:complexContent>
    <xs:extension base="tServiceSettings">
      <xs:sequence>
        <xs:choice maxOccurs="unbounded">
          <xs:element name="SmpRate">
            <xs:simpleType>
              <xs:restriction base="xs:unsignedInt">
                <xs:minExclusive value="0"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:element>
          <xs:element name="SamplesPerSec">
            <xs:simpleType>
              <xs:restriction base="xs:unsignedInt">
                <xs:minExclusive value="0"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:element>
        </xs:choice>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

```
        </xs:simpleType>
      </xs:element>
      <xs:element name="SecPerSamples">
        <xs:simpleType>
          <xs:restriction base="xs:unsignedInt">
            <xs:minExclusive value="0"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:choice>
    <xs:element name="McSecurity" type="scl:tMcSecurity" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="svID" type="tServiceSettingsEnum" use="optional" default="Fix"/>
  <xs:attribute name="optFields" type="tServiceSettingsEnum" use="optional" default="Fix"/>
  <xs:attribute name="smpRate" type="tServiceSettingsEnum" use="optional" default="Fix"/>
  <xs:attribute name="samplesPerSec" type="xs:boolean" use="optional" default="false"/>
  <xs:attribute name="pdCTimeStamp" type="xs:boolean" use="optional" default="false"/>
  <xs:attribute name="synchSrcId" type="xs:boolean" use="optional" default="false"/>
  <xs:attribute name="nofASDU" type="tServiceSettingsNoDynEnum" use="optional" default="Fix"/>
  <xs:attribute name="kdaParticipant" type="xs:boolean" use="optional" default="false"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tConflNs">
  <xs:attribute name="fixPrefix" type="xs:boolean" use="optional" default="false"/>
  <xs:attribute name="fixLnInst" type="xs:boolean" use="optional" default="false"/>
</xs:complexType>
<xs:complexType name="tValueHandling">
  <xs:attribute name="setToRO" type="xs:boolean" use="optional" default="false"/>
</xs:complexType>
<xs:complexType name="tFileHandling">
  <xs:complexContent>
    <xs:extension base="tServiceYesNo">
      <xs:attribute name="mms" type="xs:boolean" use="optional" default="true"/>
      <xs:attribute name="ftp" type="xs:boolean" use="optional" default="false"/>
      <xs:attribute name="ftps" type="xs:boolean" use="optional" default="false"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tGOOSECapabilities">
  <xs:complexContent>
    <xs:extension base="tServiceWithMax">
      <xs:attribute name="fixedOffs" type="xs:boolean" use="optional" default="false"/>
      <xs:attribute name="goose" type="xs:boolean" use="optional" default="true"/>
      <xs:attribute name="rGOOSE" type="xs:boolean" use="optional" default="false"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tRedProt">
  <xs:attribute name="hsr" type="xs:boolean" use="optional" default="false"/>
  <xs:attribute name="prp" type="xs:boolean" use="optional" default="false"/>
  <xs:attribute name="rstp" type="xs:boolean" use="optional" default="false"/>
</xs:complexType>
<xs:complexType name="tTimeSyncProt">
  <xs:complexContent>
    <xs:extension base="tServiceYesNo">
      <xs:attribute name="sntp" type="xs:boolean" use="optional" default="true"/>
      <xs:attribute name="iec61850_9_3" type="xs:boolean" use="optional" default="false"/>
      <xs:attribute name="c37_238" type="xs:boolean" use="optional" default="false"/>
      <xs:attribute name="other" type="xs:boolean" use="optional" default="false"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tSMVsc">
  <xs:complexContent>
    <xs:extension base="tServiceWithMax">
      <xs:attribute name="delivery" type="tSMVDeliveryEnum" use="optional" default="multicast"/>
      <xs:attribute name="deliveryConf" type="xs:boolean" use="optional" default="false"/>
      <xs:attribute name="sv" type="xs:boolean" use="optional" default="true"/>
      <xs:attribute name="rSV" type="xs:boolean" use="optional" default="false"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```



```

<xs:complexType name="tSupSubscription">
  <xs:attribute name="maxGo" type="xs:unsignedInt" use="required"/>
  <xs:attribute name="maxSv" type="xs:unsignedInt" use="required"/>
</xs:complexType>
<xs:complexType name="tCommProt">
  <xs:complexContent>
    <xs:extension base="tServiceYesNo">
      <xs:attribute name="ipv6" type="xs:boolean" use="optional" default="false"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tMcSecurity">
  <xs:attribute name="signature" type="xs:boolean" use="optional" default="false"/>
  <xs:attribute name="encryption" type="xs:boolean" use="optional" default="false"/>
</xs:complexType>
<xs:complexType name="tKDC">
  <xs:attribute name="iedName" type="tIEDName" use="required"/>
  <xs:attribute name="apName" type="tAccessPointName" use="required"/>
</xs:complexType>
<xs:complexType name="tSettingGroups">
  <xs:all>
    <xs:element name="SGEdit" minOccurs="0">
      <xs:complexType>
        <xs:complexContent>
          <xs:extension base="tServiceYesNo">
            <xs:attribute name="resvTms" type="xs:boolean" use="optional" default="false"/>
          </xs:extension>
        </xs:complexContent>
      </xs:complexType>
    </xs:element>
    <xs:element name="ConfSG" minOccurs="0">
      <xs:complexType>
        <xs:complexContent>
          <xs:extension base="tServiceYesNo">
            <xs:attribute name="resvTms" type="xs:boolean" use="optional" default="false"/>
          </xs:extension>
        </xs:complexContent>
      </xs:complexType>
    </xs:element>
  </xs:all>
</xs:complexType>
<xs:element name="IED" type="tIED">
  <xs:key name="LDeviceInIEDKey">
    <xs:selector xpath="/scl:AccessPoint/scl:Server/scl:LDevice"/>
    <xs:field xpath="@inst"/>
  </xs:key>
  <xs:keyref name="ref2LDeviceInDataSetForFCDAinLN" refer="LDeviceInIEDKey">
    <xs:selector xpath="/scl:AccessPoint/scl:Server/scl:LDevice/scl:LN/scl:DataSet/scl:FCDA"/>
    <xs:field xpath="@IdInst"/>
  </xs:keyref>
  <xs:keyref name="ref2LDeviceInDataSetForFCDAinLN0" refer="LDeviceInIEDKey">
    <xs:selector xpath="/scl:AccessPoint/scl:Server/scl:LDevice/scl:LN0/scl:DataSet/scl:FCDA"/>
    <xs:field xpath="@IdInst"/>
  </xs:keyref>
  <xs:key name="AccessPointInIEDKey">
    <xs:selector xpath="/scl:AccessPoint"/>
    <xs:field xpath="@name"/>
  </xs:key>
  <xs:keyref name="ServerAtRef2AccessPoint" refer="AccessPointInIEDKey">
    <xs:selector xpath="/scl:AccessPoint/scl:ServerAt"/>
    <xs:field xpath="@apName"/>
  </xs:keyref>
</xs:element>
</xs:schema>

```

<CODE ENDS>

## A.5 Communication subnetworks

### File SCL\_Communication.xsd

#### <CODE BEGINS>

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:scl="http://www.iec.ch/61850/2003/SCL"
xmlns="http://www.iec.ch/61850/2003/SCL" targetNamespace="http://www.iec.ch/61850/2003/SCL"
elementFormDefault="qualified" attributeFormDefault="unqualified" version="2007B4">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      SCL schema version "2007" revision "B" release 4, for IEC 61850-6 Ed. 2.1.

      COPYRIGHT (c) IEC, 2016. All rights reserved. Disclaimer: The IEC disclaims liability for any personal injury,
      property or other damages of any nature whatsoever, whether special, indirect, consequential or compensatory, directly or
      indirectly resulting from this software and the document upon which its methods are based, use of, or reliance upon.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="SCL_BaseTypes.xsd"/>
  <xs:complexType name="tControlBlock" abstract="true">
    <xs:complexContent>
      <xs:extension base="tUnNaming">
        <xs:sequence>
          <xs:element name="Address" type="tAddress" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="IdInst" type="tLDInst" use="required"/>
        <xs:attribute name="cbName" type="tCBName" use="required"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tCommunication">
    <xs:complexContent>
      <xs:extension base="tUnNaming">
        <xs:sequence>
          <xs:element name="SubNetwork" type="tSubNetwork" maxOccurs="unbounded">
            <xs:unique name="uniqueConnectedAP">
              <xs:selector xpath="/scl:ConnectedAP"/>
              <xs:field xpath="@iedName"/>
              <xs:field xpath="@apName"/>
            </xs:unique>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="tSubNetwork">
    <xs:complexContent>
      <xs:extension base="tNaming">
        <xs:sequence>
          <xs:element name="BitRate" type="tBitRateInMbPerSec" minOccurs="0"/>
          <xs:element name="ConnectedAP" type="tConnectedAP" maxOccurs="unbounded">
            <xs:unique name="uniqueGSEinConnectedAP">
              <xs:selector xpath="/scl:GSE"/>
              <xs:field xpath="@cbName"/>
              <xs:field xpath="@IdInst"/>
            </xs:unique>
            <xs:unique name="uniqueSMVinConnectedAP">
              <xs:selector xpath="/scl:SMV"/>
              <xs:field xpath="@cbName"/>
              <xs:field xpath="@IdInst"/>
            </xs:unique>
          </xs:element>
        </xs:sequence>
        <xs:attribute name="type" use="optional">
          <xs:simpleType>
            <xs:restriction base="xs:normalizedString">
              <xs:minLength value="1"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

</xs:complexContent>
</xs:complexType>
<xs:complexType name="tConnectedAP">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:sequence>
        <xs:element name="Address" type="tAddress" minOccurs="0"/>
        <xs:element name="GSE" type="tGSE" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="SMV" type="tSMV" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="PhysConn" type="tPhysConn" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniquePTypeInPhysConn">
            <xs:selector xpath="./scl:P"/>
            <xs:field xpath="@type"/>
          </xs:unique>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="iedName" type="tIEDName" use="required"/>
      <xs:attribute name="apName" type="tAccessPointName" use="required"/>
      <xs:attribute name="redProt" type="scl:tRedProtEnum" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tAddress">
  <xs:sequence>
    <xs:element name="P" type="tP" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="tGSE">
  <xs:complexContent>
    <xs:extension base="tControlBlock">
      <xs:sequence>
        <xs:element name="MinTime" type="tDurationInMilliSec" minOccurs="0"/>
        <xs:element name="MaxTime" type="tDurationInMilliSec" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tSMV">
  <xs:complexContent>
    <xs:extension base="tControlBlock"/>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tPhysConn">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:sequence>
        <xs:element name="P" type="tP_PhysConn" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="type" type="tPhysConnTypeEnum" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tP_PhysConn">
  <xs:simpleContent>
    <xs:extension base="tPAddr">
      <xs:attribute name="type" type="tPTypePhysConnEnum" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP">
  <xs:simpleContent>
    <xs:extension base="tPAddr">
      <xs:attribute name="type" type="tPTypeEnum" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_IPbase" abstract="true">
  <xs:simpleContent>
    <xs:restriction base="tP">
      <xs:pattern value="([0-9]{1,2}|1[0-9]{2}|2[0-4][0-9]|25[0-5])\.([0-9]{1,2}|1[0-9]{2}|2[0-4][0-9]|25[0-5])\.([0-9]{1,2}|1[0-9]{2}|2[0-4][0-9]|25[0-5])\.([0-9]{1,2}|1[0-9]{2}|2[0-4][0-9]|25[0-5])" id="IPv4"/>
    </xs:restriction>
  </xs:simpleContent>

```

```

</xs:complexType>
<xs:complexType name="tP_IP">
  <xs:simpleContent>
    <xs:restriction base="tP_IPbase">
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="IP"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_IP-SUBNET">
  <xs:simpleContent>
    <xs:restriction base="tP_IPbase">
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="IP-SUBNET"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_IP-GATEWAY">
  <xs:simpleContent>
    <xs:restriction base="tP_IPbase">
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="IP-GATEWAY"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_IPv6base" abstract="true">
  <xs:simpleContent>
    <xs:restriction base="tP">
      <xs:pattern value="([0-9a-f]{1,4};){7}[0-9a-f]{1,4}" id="IPv6"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_IPv6">
  <xs:simpleContent>
    <xs:restriction base="tP_IPv6base">
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="IPv6"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_IPv6-SUBNET">
  <xs:simpleContent>
    <xs:restriction base="tP">
      <xs:pattern value="[1-9]/[1-9][0-9]/1[0-1][0-9]/12[0-7]" id="IPv6_Subnet"/>
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="IPv6-SUBNET"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_IPv6-GATEWAY">
  <xs:simpleContent>
    <xs:restriction base="tP_IPv6base">
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="IPv6-GATEWAY"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_DNSName">
  <xs:simpleContent>
    <xs:restriction base="tP">
      <xs:pattern value="\S*"/>
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="DNSName"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_IPv6FlowLabel">
  <xs:simpleContent>
    <xs:restriction base="tP">
      <xs:pattern value="[0-9a-fA-F]{1,5}"/>
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="IPv6FlowLabel"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_OSI-NSAP">
  <xs:simpleContent>
    <xs:restriction base="tP">
      <xs:maxLength value="40"/>
      <xs:pattern value="[0-9A-F]+"/>
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="OSI-NSAP"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>

```

```

        </xs:restriction>
    </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_OSI-TSEL">
    <xs:simpleContent>
        <xs:restriction base="tP">
            <xs:maxLength value="8"/>
            <xs:pattern value="[0-9A-F]+"/>
            <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="OSI-TSEL"/>
        </xs:restriction>
    </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_OSI-SSEL">
    <xs:simpleContent>
        <xs:restriction base="tP">
            <xs:maxLength value="16"/>
            <xs:pattern value="[0-9A-F]+"/>
            <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="OSI-SSEL"/>
        </xs:restriction>
    </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_OSI-PSEL">
    <xs:simpleContent>
        <xs:restriction base="tP">
            <xs:maxLength value="16"/>
            <xs:pattern value="[0-9A-F]+"/>
            <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="OSI-PSEL"/>
        </xs:restriction>
    </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_OSI-AP-Title">
    <xs:simpleContent>
        <xs:restriction base="tP">
            <xs:pattern value="[0-9&#44;]+"/>
            <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="OSI-AP-Title"/>
        </xs:restriction>
    </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_OSI-AP-Invoke">
    <xs:simpleContent>
        <xs:restriction base="tP">
            <xs:maxLength value="5"/>
            <xs:pattern value="[0-9]+"/>
            <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="OSI-AP-Invoke"/>
        </xs:restriction>
    </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_OSI-AE-Qualifier">
    <xs:simpleContent>
        <xs:restriction base="tP">
            <xs:maxLength value="5"/>
            <xs:pattern value="[0-9]+"/>
            <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="OSI-AE-Qualifier"/>
        </xs:restriction>
    </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_OSI-AE-Invoke">
    <xs:simpleContent>
        <xs:restriction base="tP">
            <xs:maxLength value="5"/>
            <xs:pattern value="[0-9]+"/>
            <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="OSI-AE-Invoke"/>
        </xs:restriction>
    </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_MAC-Address">
    <xs:simpleContent>
        <xs:restriction base="tP">
            <xs:pattern value="[0-9A-F]{2}\-[0-9A-F]{2}\-[0-9A-F]{2}\-[0-9A-F]{2}\-[0-9A-F]{2}"/>
            <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="MAC-Address"/>
        </xs:restriction>
    </xs:simpleContent>
</xs:complexType>

```

```

<xs:complexType name="tP_APPID">
  <xs:simpleContent>
    <xs:restriction base="tP">
      <xs:pattern value="[0-9A-F]{4}"/>
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="APPID"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_VLAN-PRIORITY">
  <xs:simpleContent>
    <xs:restriction base="tP">
      <xs:pattern value="[0-7]"/>
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="VLAN-PRIORITY"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_VLAN-ID">
  <xs:simpleContent>
    <xs:restriction base="tP">
      <xs:pattern value="[0-9A-F]{3}"/>
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="VLAN-ID"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_Port" abstract="true">
  <xs:simpleContent>
    <xs:restriction base="tP">
      <xs:pattern value="0"/>
      <xs:pattern value="[1-9][0-9]{0,3}"/>
      <xs:pattern value="[1-5][0-9]{4,4}"/>
      <xs:pattern value="6[0-4][0-9]{3,3}"/>
      <xs:pattern value="65[0-4][0-9]{2,2}"/>
      <xs:pattern value="655[0-2][0-9]"/>
      <xs:pattern value="6553[0-5]"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_Sntp-Port">
  <xs:simpleContent>
    <xs:restriction base="tP_Port">
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="SNTP-Port"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_MMS-Port">
  <xs:simpleContent>
    <xs:restriction base="tP_Port">
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="MMS-Port"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_UDP-Port">
  <xs:simpleContent>
    <xs:restriction base="tP_Port">
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="IP-UDP-PORTR"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_TCP-Port">
  <xs:simpleContent>
    <xs:restriction base="tP_Port">
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="IP-TCP-PORTR"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_IPv6ClassOfTraffic">
  <xs:simpleContent>
    <xs:restriction base="tP">
      <xs:pattern value="[0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]" id="Values0-255"/>
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="IPv6ClassOfTraffic"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>

```

```

<xs:complexType name="tP_C37-118-IP-Port">
  <xs:simpleContent>
    <xs:restriction base="tP">
      <xs:pattern value="102[5-9]|10[3-9][0-9]|1[1-9][0-9][0-9][2-9][0-9]{3}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5]" id="Values1025-65535"/>
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="C37-118-IP-Port"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_IPv6-IGMPv3Src">
  <xs:simpleContent>
    <xs:restriction base="tP_IPv6base">
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="IPv6-IGMPv3Src"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_IP-IGMPv3Src">
  <xs:simpleContent>
    <xs:restriction base="tP_IPbase">
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="IP-IGMPv3Src"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tP_IP-ClassOfTraffic">
  <xs:simpleContent>
    <xs:restriction base="tP">
      <xs:maxLength value="2"/>
      <xs:pattern value="[0-9A-F]"/>
      <xs:attribute name="type" type="tPTypeEnum" use="required" fixed="IP-ClassOfTraffic"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
<xs:element name="Communication" type="tCommunication">
  <xs:unique name="uniqueSubNetwork">
    <xs:selector xpath="/scl:SubNetwork"/>
    <xs:field xpath="@name"/>
  </xs:unique>
</xs:element>
</xs:schema>

```

<CODE ENDS>

IECNORM.COM : Click to view the full PDF of IEC 61850-6:2009/AMD1:2018

## A.6 Main SCL

### File SCL.xsd

#### <CODE BEGINS>

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:scl="http://www.iec.ch/61850/2003/SCL" xmlns="http://www.iec.ch/61850/2003/SCL"
xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="http://www.iec.ch/61850/2003/SCL"
elementFormDefault="qualified" attributeFormDefault="unqualified" finalDefault="extension" version="2007B4">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      SCL schema version "2007" revision "B" release 4, for IEC 61850-6 Ed. 2.1.
      Supersedes "2007B3".

      COPYRIGHT (c) IEC, 2016. All rights reserved. Disclaimer: The IEC disclaims liability for any personal injury,
      property or other damages of any nature whatsoever, whether special, indirect, consequential or compensatory, directly or
      indirectly resulting from this software and the document upon which its methods are based, use of, or reliance upon.

      Implemented Ed. 2 Tissues (since "2007B"): 948, 1050, 1175, 1189, 1208, 1328, 1359, 1365, 1397, 1434, 1448,
      1450, 1458, 1472.
      Tissues not relevant for the SCL schema (since "2007B"): 706 (Ed.3), 837, 847, 865, 873, 883, 884, 885, 938, 949,
      961, 1048, 1054, 1059, 1118, 1130, 1131, 1147, 1161, 1168, 1170 (Ed.3), 1173, 1185, 1188, 1195, 1200, 1204,
      1207, 1221, 1224, 1241 (Ed.3), 1255, 1257 (Ed.3), 1280, 1284, 1327, 1337, 1354, 1395, 1398, 1399, 1400, 1401,
      1402, 1415, 1416, 1419, 1421, 1431, 1444, 1445, 1446, 1447, 1451, 1452, 1457, 1461, 1471.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="SCL_Substation.xsd"/>
  <xs:include schemaLocation="SCL_IED.xsd"/>
  <xs:include schemaLocation="SCL_Communication.xsd"/>
  <xs:include schemaLocation="SCL_DataTypeTemplates.xsd"/>
  <xs:element name="SCL">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="tBaseElement">
          <xs:sequence>
            <xs:element name="Header" type="tHeader">
              <xs:unique name="uniqueHitem">
                <xs:selector xpath="/scl:History/scl:Hitem"/>
                <xs:field xpath="@version"/>
                <xs:field xpath="@revision"/>
              </xs:unique>
            </xs:element>
            <xs:element ref="Substation" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element ref="Communication" minOccurs="0"/>
            <xs:element ref="IED" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element ref="DataTypeTemplates" minOccurs="0"/>
            <xs:element ref="Line" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element ref="Process" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="version" type="tScIVersion" use="required" fixed="2007"/>
          <xs:attribute name="revision" type="tScIRevision" use="required" fixed="B"/>
          <xs:attribute name="release" type="tScIRelease" use="required" fixed="4"/>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
    <xs:key name="SubstationKey">
      <xs:selector xpath="/scl:Substation|/scl:Process|/scl:Line"/>
      <xs:field xpath="@name"/>
    </xs:key>
    <xs:key name="IEDKey">
      <xs:selector xpath="/scl:IED"/>
      <xs:field xpath="@name"/>
    </xs:key>
    <xs:key name="LNodeTypeKey">
      <xs:selector xpath="/scl:DataTypeTemplates/scl:LNodeType"/>
      <xs:field xpath="@id"/>
      <xs:field xpath="@lnClass"/>
    </xs:key>
    <xs:keyref name="ref2LNodeTypeDomain1" refer="LNodeTypeKey">
      <xs:selector xpath="/scl:IED/scl:AccessPoint/scl:LN"/>
      <xs:field xpath="@lnType"/>
    </xs:keyref>
  </xs:element>

```

```
<xs:field xpath="@InClass"/>
</xs:keyref>
<xs:keyref name="ref2LNNodeTypeDomain2" refer="LNNodeTypeKey">
  <xs:selector xpath="/scl:IED/scl:AccessPoint/scl:Server/scl:LDevice/scl:LN"/>
  <xs:field xpath="@InType"/>
  <xs:field xpath="@InClass"/>
</xs:keyref>
<xs:keyref name="ref2LNNodeTypeLLN0" refer="LNNodeTypeKey">
  <xs:selector xpath="/scl:IED/scl:AccessPoint/scl:Server/scl:LDevice/scl:LN0"/>
  <xs:field xpath="@InType"/>
  <xs:field xpath="@InClass"/>
</xs:keyref>
<xs:keyref name="refConnectedAP2IED" refer="IEDKey">
  <xs:selector xpath="/scl:Communication/scl:SubNetwork/scl:ConnectedAP"/>
  <xs:field xpath="@iedName"/>
</xs:keyref>
<xs:keyref name="ref2SubstationFromTerminal" refer="scl:SubstationKey">
  <xs:selector xpath="//scl:Terminal"/>
  <xs:field xpath="@substationName"/>
</xs:keyref>
<xs:key name="ConnectivityNodeKey">
  <xs:selector xpath="//scl:ConnectivityNode"/>
  <xs:field xpath="@pathName"/>
</xs:key>
</xs:element>
</xs:schema>
```

**<CODE ENDS>**

IECNORM.COM : Click to view the full PDF of IEC 61850-6:2009/AMD1:2018

**Annex B**  
(informative)

**SCL enumerations according to IEC 61850-7-3 and IEC 61850-7-4**

*Replace the contents of this Annex with the following text:*

The enumeration definitions related to IEC 61850-7-2, -7-3 and -7-4 can be found in the respective second editions.

IECNORM.COM : Click to view the full PDF of IEC 61850-6:2009/AMD1:2018

**Annex C**  
(informative)

**Syntax extension examples**

**C.2 Extension syntax for data model description**

*Remove existing Clause C.2, which is replaced by part 7-7.*

IECNORM.COM : Click to view the full PDF of IEC 61850-6:2009/AMD1:2018

## Annex D (informative)

### D.2 Example SCL file contents

Replace the existing schema in Subclause D.2 with the following new schema:

```
<?xml version="1.0"?>
<SCL xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.iec.ch/61850/2003/SCL" version="2007"
revision="B" release="3" xsi:schemaLocation="http://www.iec.ch/61850/2003/SCL
file:///C:/Data/SCLXSD/SCL.2007B4/SCL.xsd">
  <Header id="SCL Example T1-1" toolID="MySystemTool"/>
  <Substation name="S12" desc="Baden">
    <PowerTransformer name="T1" type="PTR">
      <LNNode iedName="D1" InClass="PDIF" IdInst="F1" iedName="D1Q1BP2"/>
      <LNNode iedName="D1" InClass="YLTC" IdInst="S12D1T1" iedName="None"/>
      <TransformerWinding name="W1" type="PTW">
        <Terminal connectivityNode="S12/D1/Q1/L1" substationName="S12" voltageLevelName="D1" bayName="Q1"
cNodeName="L1"/>
      </TransformerWinding>
      <TransformerWinding name="W2" type="PTW">
        <Terminal connectivityNode="S12/E1/Q2/L3" substationName="S12" voltageLevelName="E1" bayName="Q2"
cNodeName="L3"/>
      </TransformerWinding>
    </PowerTransformer>
    <VoltageLevel name="D1">
      <Voltage multiplier="k" unit="V">220</Voltage>
      <Bay name="Q1">
        <LNNode iedName="None" IdInst="S12D1Q1" InClass="PDIS" InInst="1"/>
        <ConductingEquipment name="I1" type="CTR">
          <Terminal connectivityNode="S12/D1/Q1/L1" substationName="S12" voltageLevelName="D1"
bayName="Q1" cNodeName="L1"/>
          <SubEquipment name="R" phase="A">
            <LNNode iedName="D1Q1BP2" IdInst="F1" InClass="TCTR" InInst="1"/>
          </SubEquipment>
          <SubEquipment name="S" phase="B">
            <LNNode iedName="D1Q1BP2" IdInst="F1" InClass="TCTR" InInst="2"/>
          </SubEquipment>
          <SubEquipment name="T" phase="C">
            <LNNode iedName="D1Q1BP2" IdInst="F1" InClass="TCTR" InInst="3"/>
          </SubEquipment>
          <SubEquipment name="IO" phase="N">
            <LNNode iedName="D1Q1BP2" IdInst="F1" InClass="TCTR" InInst="4"/>
          </SubEquipment>
        </ConductingEquipment>
        <ConnectivityNode name="L1" pathName="S12/D1/Q1/L1"/>
      </Bay>
    </VoltageLevel>
    <VoltageLevel name="E1">
      <Voltage multiplier="k" unit="V">132</Voltage>
      <Bay name="Q1">
        <LNNode iedName="E1Q1SB1" IdInst="C1" InClass="MMXN" InInst="1"/>
        <LNNode iedName="None" IdInst="S12E1Q1" InClass="PDIS" InInst="1"/>
        <LNNode iedName="None" IdInst="F1" InClass="PDIF" InInst="1"/>
        <ConductingEquipment name="QA1" type="CBR">
          <LNNode iedName="E1Q1SB1" IdInst="C1" InClass="CSWI" InInst="1"/>
          <Terminal connectivityNode="S12/E1/Q1/L1" substationName="S12" voltageLevelName="E1"
bayName="Q1" cNodeName="L1"/>
          <Terminal connectivityNode="S12/E1/Q1/L2" substationName="S12" voltageLevelName="E1"
bayName="Q1" cNodeName="L2"/>
        </ConductingEquipment>
        <ConductingEquipment name="QB1" type="DIS">
          <LNNode iedName="E1Q1SB1" IdInst="C1" InClass="CSWI" InInst="2"/>
          <LNNode iedName="D1Q1SB4" IdInst="C1" InClass="CILO" InInst="1"/>
          <Terminal connectivityNode="S12/E1/Q4/B1" substationName="S12" voltageLevelName="E1"
bayName="Q4" cNodeName="B1"/>
          <Terminal connectivityNode="S12/E1/Q1/L1" substationName="S12" voltageLevelName="E1"
bayName="Q1" cNodeName="L1"/>
        </ConductingEquipment>
      </Bay>
    </VoltageLevel>
  </Substation>
</SCL>
```

```

    <ConductingEquipment name="U1" type="VTR">
      <Terminal connectivityNode="S12/E1/Q1/L2" substationName="S12" voltageLevelName="E1"
bayName="Q1" cNodeName="L2"/>
      <SubEquipment name="A" phase="A">
        <LNode iedName="E1Q1SB1" IdInst="C1" InClass="TVTR" InInst="1"/>
      </SubEquipment>
    </ConductingEquipment>
    <ConductingEquipment name="I1" type="CTR">
      <Terminal connectivityNode="S12/E1/Q1/L3" substationName="S12" voltageLevelName="E1"
bayName="Q1" cNodeName="L3"/>
      <Terminal connectivityNode="S12/E1/Q1/L2" substationName="S12" voltageLevelName="E1"
bayName="Q1" cNodeName="L2"/>
    </ConductingEquipment>
    <ConnectivityNode name="L1" pathName="S12/E1/Q1/L1"/>
    <ConnectivityNode name="L2" pathName="S12/E1/Q1/L2"/>
    <ConnectivityNode name="L3" pathName="S12/E1/Q1/L3"/>
  </Bay>
  <Bay name="Q2" desc="Turgi">
    <ConductingEquipment name="QA1" type="CBR">
      <LNode iedName="D1Q1SB4" IdInst="C1" InClass="CSWI" InInst="1"/>
      <Terminal connectivityNode="S12/E1/Q2/L0" substationName="S12" voltageLevelName="E1"
bayName="Q2" cNodeName="L0"/>
      <Terminal connectivityNode="S12/E1/Q2/L1" substationName="S12" voltageLevelName="E1"
bayName="Q2" cNodeName="L1"/>
    </ConductingEquipment>
    <ConductingEquipment name="QB1" type="DIS">
      <LNode iedName="D1Q1SB4" IdInst="C1" InClass="CSWI" InInst="2"/>
      <LNode iedName="D1Q1SB4" IdInst="C1" InClass="CILO" InInst="2"/>
      <Terminal connectivityNode="S12/E1/Q4/B1" substationName="S12" voltageLevelName="E1"
bayName="Q4" cNodeName="B1"/>
      <Terminal connectivityNode="S12/E1/Q2/L0" substationName="S12" voltageLevelName="E1"
bayName="Q2" cNodeName="L0"/>
    </ConductingEquipment>
    <ConductingEquipment name="I1" type="CTR">
      <Terminal connectivityNode="S12/E1/Q2/L1" substationName="S12" voltageLevelName="E1"
bayName="Q2" cNodeName="L1"/>
      <Terminal connectivityNode="S12/E1/Q2/L2" substationName="S12" voltageLevelName="E1"
bayName="Q2" cNodeName="L2"/>
    </ConductingEquipment>
    <ConductingEquipment name="U1" type="VTR">
      <Terminal connectivityNode="S12/E1/Q2/L1" substationName="S12" voltageLevelName="E1"
bayName="Q2" cNodeName="L1"/>
    </ConductingEquipment>
    <ConnectivityNode name="L0" pathName="S12/E1/Q2/L0"/>
    <ConnectivityNode name="L1" pathName="S12/E1/Q2/L1"/>
    <ConnectivityNode name="L2" pathName="S12/E1/Q2/L2"/>
    <ConnectivityNode name="L3" pathName="S12/E1/Q2/L3"/>
  </Bay>
  <Bay name="Q3" desc="London">
    <LNode iedName="None" IdInst="LD0" InClass="MMXN" InInst="1"/>
    <LNode iedName="None" IdInst="LD0" InClass="PDIS" InInst="1"/>
    <LNode iedName="None" IdInst="LD0" InClass="PDIF" InInst="1"/>
    <ConductingEquipment name="QA1" type="CBR">
      <LNode iedName="None" IdInst="C1" InClass="CSWI" InInst="1"/>
      <Terminal connectivityNode="S12/E1/Q3/L1" substationName="S12" voltageLevelName="E1"
bayName="Q3" cNodeName="L1"/>
      <Terminal connectivityNode="S12/E1/Q3/L2" substationName="S12" voltageLevelName="E1"
bayName="Q3" cNodeName="L2"/>
    </ConductingEquipment>
    <ConductingEquipment name="QB1" type="DIS">
      <Terminal connectivityNode="S12/E1/Q4/B1" substationName="S12" voltageLevelName="E1"
bayName="Q4" cNodeName="B1"/>
      <Terminal connectivityNode="S12/E1/Q3/L1" substationName="S12" voltageLevelName="E1"
bayName="Q3" cNodeName="L1"/>
    </ConductingEquipment>
    <ConductingEquipment name="U1" type="VTR">
      <Terminal connectivityNode="S12/E1/Q3/L2" substationName="S12" voltageLevelName="E1"
bayName="Q3" cNodeName="L2"/>
    </ConductingEquipment>
    <ConductingEquipment name="I1" type="CTR">
      <Terminal connectivityNode="S12/E1/Q3/L3" substationName="S12" voltageLevelName="E1"
bayName="Q3" cNodeName="L3"/>
  </Bay>

```

```
<Terminal connectivityNode="S12/E1/Q3/L2" substationName="S12" voltageLevelName="E1"
bayName="Q3" cNodeName="L2"/>
</ConductingEquipment>
<ConnectivityNode name="L1" pathName="S12/E1/Q3/L1"/>
<ConnectivityNode name="L2" pathName="S12/E1/Q3/L2"/>
<ConnectivityNode name="L3" pathName="S12/E1/Q3/L3"/>
</Bay>
<Bay name="Q4">
  <ConnectivityNode name="B1" pathName="S12/E1/Q4/B1"/>
</Bay>
</VoltageLevel>
</Substation>
<Communication>
  <SubNetwork name="W01" type="8-MMS">
    <Text>Station bus</Text>
    <BitRate unit="b/s">100</BitRate>
    <ConnectedAP iedName="D1Q1SB4" apName="S1">
      <Address>
        <P type="IP">10.0.0.11</P>
        <P type="IP-SUBNET">255.255.255.0</P>
        <P type="IP-GATEWAY">10.0.0.101</P>
        <P type="OSI-TSEL">00000001</P>
        <P type="OSI-PSEL">01</P>
        <P type="OSI-SSEL">01</P>
      </Address>
      <GSE IdInst="C1" cbName="SyckResult">
        <Address>
          <P type="MAC-Address">01-0C-CD-01-00-02</P>
          <P type="APPID">3001</P>
          <P type="VLAN-PRIORITY">4</P>
        </Address>
        <MinTime unit="s">4</MinTime>
        <MaxTime unit="s">1000</MaxTime>
      </GSE>
      <PhysConn type="Connection">
        <P type="Type">FOC</P>
        <P type="Plug">ST</P>
      </PhysConn>
    </ConnectedAP>
    <ConnectedAP iedName="E1Q1SB1" apName="S1">
      <Address>
        <P type="IP">10.0.0.1</P>
        <P type="IP-SUBNET">255.255.255.0</P>
        <P type="IP-GATEWAY">10.0.0.101</P>
        <P type="OSI-TSEL">00000001</P>
        <P type="OSI-PSEL">01</P>
        <P type="OSI-SSEL">01</P>
      </Address>
      <GSE IdInst="C1" cbName="ItIPositions">
        <Address>
          <P type="MAC-Address">01-0C-CD-01-00-01</P>
          <P type="APPID">3000</P>
          <P type="VLAN-PRIORITY">4</P>
        </Address>
      </GSE>
      <SMV IdInst="C1" cbName="Volt">
        <Address>
          <P type="MAC-Address">01-0C-CD-04-00-01</P>
          <P type="APPID">4000</P>
          <P type="VLAN-ID">123</P>
          <P type="VLAN-PRIORITY">4</P>
        </Address>
      </SMV>
    </ConnectedAP>
    <ConnectedAP iedName="E1Q1BP2" apName="S1">
      <Address>
        <P type="IP">10.0.0.2</P>
        <P type="IP-SUBNET">255.255.255.0</P>
        <P type="IP-GATEWAY">10.0.0.101</P>
        <P type="OSI-TSEL">00000001</P>
        <P type="OSI-PSEL">01</P>
        <P type="OSI-SSEL">01</P>
      </Address>
```

IECNORM.COM - View the full PDF of IEC 61850-6:2009/AMD1:2018

```

</ConnectedAP>
<ConnectedAP iedName="E1Q1BP3" apName="S1">
  <Address>
    <P type="IP">10.0.0.3</P>
    <P type="IP-SUBNET">255.255.255.0</P>
    <P type="IP-GATEWAY">10.0.0.101</P>
    <P type="OSI-TSEL">00000001</P>
    <P type="OSI-PSEL">01</P>
    <P type="OSI-SSEL">01</P>
  </Address>
</ConnectedAP>
<ConnectedAP iedName="A1KA1" apName="S1">
  <Address>
    <P type="IP">10.0.0.121</P>
    <P type="IP-SUBNET">255.255.255.0</P>
    <P type="IP-GATEWAY">10.0.0.101</P>
    <P type="OSI-TSEL">00000001</P>
    <P type="OSI-PSEL">01</P>
    <P type="OSI-SSEL">01</P>
  </Address>
</ConnectedAP>
<ConnectedAP iedName="E1Q2SB1" apName="S1">
  <Address>
    <P type="IP">10.0.0.14</P>
    <P type="IP-SUBNET">255.255.255.0</P>
    <P type="IP-GATEWAY">10.0.0.101</P>
    <P type="OSI-TSEL">00000001</P>
    <P type="OSI-PSEL">01</P>
    <P type="OSI-SSEL">01</P>
  </Address>
</ConnectedAP>
</SubNetwork>
</Communication>
<IED name="E1Q1SB1">
  <Services nameLength="64">
    <ClientServices goose="true" sv="true"/>
    <DynAssociation/>
    <GetDirectory/>
    <GetDataObjectDefinition/>
    <GetDataSetValue/>
    <DataSetDirectory/>
    <ConfDataSet max="4" maxAttributes="50"/>
    <ReadWrite/>
    <ConfReportControl max="12"/>
    <GetCBValues/>
    <ConfLogControl max="1"/>
    <ReportSettings cbName="Conf" datSet="Conf" rptID="Dyn" optFields="Conf" bufTime="Dyn" intgPd="Dyn"/>
    <GSESettings cbName="Conf" datSet="Conf" appID="Conf"/>
    <GOOSE max="2"/>
    <SMVSettings cbName="Conf" datSet="Conf" optFields="Fix" smpRate="Conf" svID="Conf">
      <SmpRate>80</SmpRate>
      <SmpRate>240</SmpRate>
    </SMVSettings>
    <FileHandling/>
    <ConfLNs fixLnInst="true"/>
  </Services>
  <AccessPoint name="S1">
    <Server>
      <Authentication none="true"/>
      <LDevice inst="C1">
        <LN0 inst="" InClass="LLN0" InType="LN0">
          <DataSet name="Positions">
            <FCDA IdInst="C1" prefix="" InClass="CSWI" InInst="1" doName="Pos" fc="ST"/>
            <FCDA IdInst="C1" prefix="" InClass="CSWI" InInst="2" doName="Pos" fc="ST"/>
          </DataSet>
          <DataSet name="Measurands">
            <FCDA IdInst="C1" prefix="" InClass="MMXN" InInst="1" doName="Amp" fc="MX"/>
            <FCDA IdInst="C1" prefix="" InClass="MMXN" InInst="1" doName="Vol" fc="MX"/>
          </DataSet>
          <DataSet name="smv">
            <FCDA IdInst="C1" prefix="" InClass="TVTR" InInst="1" doName="VolSv" daName="instMag"
fc="MX"/>
          </DataSet>
        </LN0>
      </LDevice>
    </Server>
  </AccessPoint>
</IED>

```

```
<ReportControl name="PosReport" rptID="E1Q1Switches" dataSet="Positions" confRev="1">
  <TrgOps dchg="true" qchg="true"/>
  <OptFields/>
  <RptEnabled max="5">
    <ClientLN iedName="A1KA1" IdInst="none" lnInst="1" lnClass="IHMI"/>
  </RptEnabled>
</ReportControl>
intgPd="2000">
  <ReportControl name="MeaReport" rptID="E1Q1Measurands" dataSet="Measurands" confRev="1">
    <TrgOps qchg="true" period="true"/>
    <OptFields reasonCode="true"/>
    <RptEnabled max="5">
      <ClientLN iedName="A1KA1" IdInst="none" lnInst="1" lnClass="IHMI"/>
    </RptEnabled>
  </ReportControl>
  <LogControl name="Log" dataSet="Positions" logName="C1">
    <TrgOps dchg="true" qchg="true"/>
  </LogControl>
  <Log/>
  <GSEControl name="ItlPositions" dataSet="Positions" appID="Itl">
    <IEDName>E1Q2SB1</IEDName>
  </GSEControl>
  <SampledValueControl name="Volt" dataSet="smv" smvID="11" smpRate="4800" nofASDU="5">
    <IEDName>D1Q1SB4</IEDName>
    <SmvOpts refreshTime="true" sampleSynchronized="true" sampleRate="true"/>
  </SampledValueControl>
</LN0>
<LN inst="1" lnClass="LPHD" lnType="LPHDa">
  <DOI name="Proxy">
    <DAI name="stVal" valKind="Set">
      <Val>false</Val>
    </DAI>
  </DOI>
</LN>
<LN inst="1" lnClass="CSWI" lnType="CSWIa"/>
<LN inst="2" lnClass="CSWI" lnType="CSWIa"/>
<LN inst="1" lnClass="MMXN" lnType="MMXNa">
  <DOI name="Vol">
    <SDI name="sVC">
      <DAI name="scaleFactor" valKind="Set">
        <Val>200</Val>
      </DAI>
      <DAI name="offset" valKind="Set">
        <Val>10</Val>
      </DAI>
    </SDI>
  </DOI>
</LN>
<LN inst="1" lnClass="TVTR" lnType="TVTRa"/>
</LDevice>
</Server>
</AccessPoint>
</IED>
<IED name="E1Q1BP2">
  <AccessPoint name="S1"/>
</IED>
<IED name="E1Q1BP3">
  <AccessPoint name="S1"/>
</IED>
<IED name="E1Q2SB1">
  <AccessPoint name="S1"/>
</IED>
<IED name="E1Q3SB1">
  <AccessPoint name="S1"/>
</IED>
<IED name="E1Q3KA1">
  <AccessPoint name="S1"/>
</IED>
<IED name="E1Q3KA2">
  <AccessPoint name="S1"/>
</IED>
<IED name="E1Q3KA3">
  <AccessPoint name="S1"/>
</IED>
```

```

</IED>
<IED name="D1Q1SB1">
  <AccessPoint name="S1"/>
</IED>
<IED name="D1Q1BP2">
  <Services nameLength="64">
    <DynAssociation/>
    <GetDirectory/>
    <GetDataObjectDefinition/>
    <GetDataSetValue/>
    <DataSetDirectory/>
    <ConfDataSet max="4"/>
    <ReadWrite/>
    <ConfReportControl max="12"/>
    <GetCBValues/>
    <ReportSettings cbName="Conf" datSet="Conf" rptID="Dyn" optFields="Dyn" bufTime="Dyn" intgPd="Dyn"/>
    <GSESettings cbName="Conf" datSet="Conf" appID="Conf"/>
    <GOOSE max="2"/>
    <FileHandling/>
  </Services>
  <AccessPoint name="S1">
    <Server>
      <Authentication none="true"/>
      <LDevice inst="F1">
        <LN0 inst="" InClass="LLN0" InType="LNO"/>
        <LN inst="1" InClass="LPHD" InType="LPHDa">
          <DOI name="Proxy">
            <DAI name="stVal" valKind="Set">
              <Val>false</Val>
            </DAI>
          </DOI>
        </LN>
        <LN inst="1" InClass="TCTR" InType="TCTRa"/>
        <LN inst="2" InClass="TCTR" InType="TCTRa"/>
        <LN inst="3" InClass="TCTR" InType="TCTRa"/>
        <LN inst="4" InClass="TCTR" InType="TCTRa"/>
        <LN InType="PDIFa" inst="1" InClass="PDIF">
          <DOI name="TmASt" desc="Example of array value definition – function wise meaningless">
            <SDI name="crvPts" ix="1">
              <DAI name="xVal">
                <Val>12.5</Val>
              </DAI>
              <DAI name="yVal">
                <Val>22.1</Val>
              </DAI>
            </SDI>
            <SDI name="crvPts" ix="2">
              <DAI name="xVal">
                <Val>102.5</Val>
              </DAI>
              <DAI name="yVal">
                <Val>202.1</Val>
              </DAI>
            </SDI>
          </DOI>
        </LN>
      </LDevice>
    </Server>
  </AccessPoint>
</IED>
<IED name="D1Q1BP3">
  <AccessPoint name="S1"/>
</IED>
<IED name="D1Q1SB4">
  <Services nameLength="64">
    <DynAssociation/>
    <GetDirectory/>
    <GetDataObjectDefinition/>
    <GetDataSetValue/>
    <DataSetDirectory/>
    <ConfDataSet max="4"/>
    <ReadWrite/>
    <ConfReportControl max="12"/>
  </Services>
</IED>

```



```

    <GetCBValues/>
    <ConfLogControl max="1"/>
    <ReportSettings cbName="Conf" dataSet="Conf" rptID="Dyn" optFields="Conf" bufTime="Dyn" intgPd="Dyn"/>
    <GSESettings cbName="Conf" dataSet="Conf" appID="Conf"/>
    <GOOSE max="2"/>
    <FileHandling/>
  </Services>
  <AccessPoint name="S1">
    <Server>
      <Authentication none="true"/>
      <LDevice inst="C1">
        <LN0 inst="" InClass="LLNO" InType="LNO">
          <DataSet name="SyckResult">
            <FCDA IdInst="C1" prefix="" InClass="RSYN" InInst="1" doName="Rel" fc="ST"/>
          </DataSet>
          <GSEControl name="SyckResult" dataSet="SyckResult" confRev="1" appID="SynChk">
            <IEDName>E1Q1SB1</IEDName>
          </GSEControl>
        </LN0>
        <LN inst="1" InClass="LPHD" InType="LPHDa">
          <DOI name="Proxy">
            <DAI name="stVal" valKind="Set">
              <Val>false</Val>
            </DAI>
          </DOI>
        </LN>
        <LN inst="1" InClass="RSYN" InType="RSYNa"/>
        <LN inst="1" InClass="CILO" InType="CILOa"/>
        <LN inst="2" InClass="CILO" InType="CILOa"/>
        <LN inst="1" InClass="CSWI" InType="CSWIa"/>
        <LN inst="2" InClass="CSWI" InType="CSWIa"/>
      </LDevice>
    </Server>
  </AccessPoint>
</IED>
<IED name="A1KA1">
  <AccessPoint name="S1">
    <LN inst="1" InClass="IHMI" InType="IHMIa"/>
  </AccessPoint>
</IED>
<DataTypeTemplates>
  <LNNodeType id="LNO" InClass="LLNO">
    <DO name="Mod" type="myMod"/>
    <DO name="Beh" type="myBeh"/>
    <DO name="Health" type="myHealth"/>
    <DO name="NamPlt" type="myLN0LPL"/>
  </LNNodeType>
  <LNNodeType id="LPHDa" InClass="LPHD">
    <DO name="PhyNam" type="myDPL"/>
    <DO name="PhyHealth" type="myHealth"/>
    <DO name="Proxy" type="mySPS"/>
  </LNNodeType>
  <LNNodeType id="CSWIa" InClass="CSWI">
    <DO name="Mod" type="myMod"/>
    <DO name="Beh" type="myBeh"/>
    <DO name="Health" type="myHealth"/>
    <DO name="NamPlt" type="myLPL"/>
    <DO name="Pos" type="myPos"/>
  </LNNodeType>
  <LNNodeType id="MMXNa" InClass="MMXN">
    <DO name="Mod" type="myMod"/>
    <DO name="Beh" type="myBeh"/>
    <DO name="Health" type="myHealth"/>
    <DO name="NamPlt" type="myLPL"/>
    <DO name="Amp" type="myMV"/>
    <DO name="Vol" type="myMV"/>
  </LNNodeType>
  <LNNodeType id="CILOa" InClass="CILO">
    <DO name="Mod" type="myMod"/>
    <DO name="Beh" type="myBeh"/>
    <DO name="Health" type="myHealth"/>
    <DO name="NamPlt" type="myLPL"/>
    <DO name="EnaOpn" type="mySPS"/>
  </LNNodeType>

```

```
<DO name="EnaCls" type="mySPS"/>
</LNodeType>
<LNodeType id="TVTRa" InClass="TVTR">
  <DO name="Mod" type="myMod"/>
  <DO name="Beh" type="myBeh"/>
  <DO name="Health" type="myHealth"/>
  <DO name="NamPlt" type="myLPL"/>
  <DO name="VolSv" type="mySAV"/>
  <DO name="FuFail" type="mySPS"/>
</LNodeType>
<LNodeType id="TCTRa" InClass="TCTR">
  <DO name="Mod" type="myMod"/>
  <DO name="Beh" type="myBeh"/>
  <DO name="Health" type="myHealth"/>
  <DO name="NamPlt" type="myLPL"/>
  <DO name="AmpSv" type="mySAV"/>
</LNodeType>
<LNodeType id="RSYNa" InClass="RSYN">
  <DO name="Mod" type="myMod"/>
  <DO name="Beh" type="myBeh"/>
  <DO name="Health" type="myHealth"/>
  <DO name="NamPlt" type="myLPL"/>
  <DO name="Rel" type="mySPS"/>
</LNodeType>
<LNodeType id="IHMIa" InClass="IHMI">
  <DO name="Mod" type="myMod"/>
  <DO name="Beh" type="myBeh"/>
  <DO name="Health" type="myHealth"/>
  <DO name="NamPlt" type="myLPL"/>
</LNodeType>
<LNodeType id="PDIFa" InClass="PDIF">
  <DO name="Mod" type="myMod"/>
  <DO name="Beh" type="myBeh"/>
  <DO name="Health" type="myHealth"/>
  <DO name="NamPlt" type="myLPL"/>
  <DO name="Str" type="myACD"/>
  <DO name="Op" type="myACT"/>
  <DO name="TmASt" type="myCSD"/>
</LNodeType>
<DOType id="myMod" cdc="ENC">
  <DA name="stVal" fc="ST" dchg="true" bType="Enum" type="BehaviourModeKind"/>
  <DA name="q" fc="ST" qchg="true" bType="Quality"/>
  <DA name="t" fc="ST" bType="Timestamp"/>
  <DA name="ctlModel" fc="CF" bType="Enum" type="CtlModelKind" dchg="true"/>
  <DA name="Oper" fc="CO" bType="Struct" type="myModOper"/>
</DOType>
<DOType id="myHealth" cdc="ENS">
  <DA name="stVal" fc="ST" dchg="true" bType="Enum" type="HealthKind"/>
  <DA name="q" fc="ST" qchg="true" bType="Quality"/>
  <DA name="t" fc="ST" bType="Timestamp"/>
</DOType>
<DOType id="myBeh" cdc="ENS">
  <DA name="stVal" fc="ST" dchg="true" bType="Enum" type="BehaviourModeKind"/>
  <DA name="q" fc="ST" qchg="true" bType="Quality"/>
  <DA name="t" fc="ST" bType="Timestamp"/>
</DOType>
<DOType id="myINS" cdc="INS">
  <DA name="stVal" fc="ST" dchg="true" bType="INT32"/>
  <DA name="q" fc="ST" qchg="true" bType="Quality"/>
  <DA name="t" fc="ST" bType="Timestamp"/>
</DOType>
<DOType id="myLN0LPL" cdc="LPL">
  <DA name="vendor" fc="DC" bType="VisString255">
    <Val>myVendorName</Val>
  </DA>
  <DA name="swRev" fc="DC" bType="VisString255">
    <Val>my SW revision ID</Val>
  </DA>
  <DA name="d" fc="DC" bType="VisString255"/>
  <DA name="configRev" fc="DC" bType="VisString255">
    <Val>Rev 3.45</Val>
  </DA>
  <DA name="IdNs" fc="EX" bType="VisString255">
```



```
<Val>IEC 61850-7-4:2007</Val>
</DA>
</DOType>
<DOType id="myLPL" cdc="LPL">
  <DA name="vendor" fc="DC" bType="VisString255">
    <Val>myVendorName</Val>
  </DA>
  <DA name="swRev" fc="DC" bType="VisString255"/>
  <DA name="d" fc="DC" bType="VisString255"/>
</DOType>
<DOType id="myDPL" cdc="DPL">
  <DA name="vendor" fc="DC" bType="VisString255">
    <Val>myVendorName</Val>
  </DA>
  <DA name="hwRev" fc="DC" bType="VisString255">
    <Val>Rev 1.23</Val>
  </DA>
</DOType>
<DOType id="myPos" cdc="DPC">
  <DA name="stVal" fc="ST" dchg="true" bType="Dbpos"/>
  <DA name="q" fc="ST" qchg="true" bType="Quality"/>
  <DA name="t" fc="ST" bType="Timestamp"/>
  <DA name="ctlModel" fc="CF" bType="Enum" type="CtlModelKind" dchg="true"/>
  <DA name="Oper" fc="CO" bType="Struct" type="myOper"/>
  <DA name="SBOW" fc="CO" bType="Struct" type="myOper"/>
  <DA name="Cancel" fc="CO" bType="Struct" type="myCancel"/>
</DOType>
<DOType id="mySPS" cdc="SPS">
  <DA name="stVal" fc="ST" dchg="true" bType="BOOLEAN"/>
  <DA name="q" fc="ST" qchg="true" bType="Quality"/>
  <DA name="t" fc="ST" bType="Timestamp"/>
</DOType>
<DOType id="myMV" cdc="MV">
  <DA name="mag" fc="MX" dchg="true" bType="Struct" type="myAnalogValue"/>
  <DA name="q" fc="MX" qchg="true" bType="Quality"/>
  <DA name="t" fc="MX" bType="Timestamp"/>
  <DA name="sVC" fc="CF" dchg="true" bType="Struct" type="ScaledValueConfig"/>
</DOType>
<DOType id="myCMV" cdc="CMV">
  <DA name="cVal" fc="MX" dchg="true" bType="Struct" type="myVector"/>
  <DA name="q" fc="MX" qchg="true" bType="Quality"/>
  <DA name="t" fc="MX" bType="Timestamp"/>
</DOType>
<DOType id="mySEQ" cdc="SEQ">
  <SDO name="c1" type="myCMV"/>
  <SDO name="c2" type="myCMV"/>
  <SDO name="c3" type="myCMV"/>
  <DA name="seqT" fc="MX" bType="Enum" type="SequenceKind"/>
</DOType>
<DOType id="myACD" cdc="ACD">
  <DA name="general" fc="ST" dchg="true" bType="BOOLEAN"/>
  <DA name="dirGeneral" fc="ST" dchg="true" bType="Enum" type="FaultDirectionKind"/>
  <DA name="q" fc="ST" qchg="true" bType="Quality"/>
  <DA name="t" fc="ST" bType="Timestamp"/>
</DOType>
<DOType id="myACT" cdc="ACT">
  <DA name="general" fc="ST" dchg="true" bType="BOOLEAN"/>
  <DA name="q" fc="ST" qchg="true" bType="Quality"/>
  <DA name="t" fc="ST" bType="Timestamp"/>
</DOType>
<DOType id="myCSD" cdc="CSD">
  <DA name="xUnits" fc="DC" bType="Struct" type="myUnit"/>
  <DA name="xD" fc="DC" bType="VisString255"/>
  <DA name="yUnits" fc="DC" bType="Struct" type="myUnit"/>
  <DA name="yD" fc="DC" bType="VisString255"/>
  <DA name="numPts" fc="DC" bType="INT16U"/>
  <DA name="crvPts" fc="DC" count="maxPts" bType="Struct" type="xyPoint"/>
  <DA name="d" fc="DC" bType="VisString255"/>
  <DA name="maxPts" fc="CF" bType="INT16U" dchg="true">
    <Val>32</Val>
  </DA>
</DOType>
<DOType id="mySAV" cdc="SAV">
```

```

    <DA name="instMag" fc="MX" bType="Struct" type="myAnalogValue"/>
    <DA name="q" fc="MX" qchg="true" bType="Quality"/>
</DOType>
<DAType id="myAnalogValue">
    <BDA name="f" bType="FLOAT32"/>
</DAType>
<DAType id="ScaledValueConfig">
    <BDA name="scaleFactor" bType="FLOAT32"/>
    <BDA name="offset" bType="FLOAT32"/>
</DAType>
<DAType id="myVector">
    <BDA name="mag" bType="Struct" type="myAnalogValue"/>
    <BDA name="ang" bType="Struct" type="myAnalogValue"/>
</DAType>
<DAType id="xyPoint">
    <BDA name="xVal" bType="FLOAT32"/>
    <BDA name="yVal" bType="FLOAT32"/>
</DAType>
<DAType id="myUnit">
    <BDA name="SIUnit" bType="Enum" sAddr="" type="SIUnitKind"/>
    <BDA name="multiplier" bType="Enum" sAddr="" type="MultiplierKind">
        <Val>k</Val>
    </BDA>
</DAType>
<DAType id="originator">
    <BDA name="orCat" bType="Enum" type="OriginatorCategoryKind"/>
    <BDA name="orIdent" bType="Octet64"/>
</DAType>
<DAType id="myModOper">
    <BDA name="ctlVal" bType="Enum" type="BehaviourModeKind"/>
    <BDA name="origin" bType="Struct" type="originator"/>
    <BDA name="ctlNum" bType="INT8U"/>
    <BDA name="T" bType="Timestamp"/>
    <BDA name="Test" bType="BOOLEAN"/>
    <BDA name="Check" bType="Check"/>
    <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
</DAType>
<DAType id="myOper">
    <BDA name="ctlVal" bType="BOOLEAN"/>
    <BDA name="origin" bType="Struct" type="originator"/>
    <BDA name="ctlNum" bType="INT8U"/>
    <BDA name="T" bType="Timestamp"/>
    <BDA name="Test" bType="BOOLEAN"/>
    <BDA name="Check" bType="Check"/>
    <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
</DAType>
<DAType id="myCancel">
    <BDA name="ctlVal" bType="BOOLEAN"/>
    <BDA name="origin" bType="Struct" type="originator"/>
    <BDA name="ctlNum" bType="INT8U"/>
    <BDA name="T" bType="Timestamp"/>
    <BDA name="Test" bType="BOOLEAN"/>
    <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
</DAType>
<EnumType id="BehaviourModeKind" desc="Behaviour or mode (BehaviourModeKind enumeration)">
    <EnumVal ord="1" desc="Normal enabled state.">on</EnumVal>
    <EnumVal ord="2" desc="Process is passively supervised.">blocked</EnumVal>
    <EnumVal ord="3" desc="Function is operated but results are indicated as test results.">test</EnumVal>
    <EnumVal ord="4" desc="Function is operated in test mode, but with no impact to the
process.">test/blocked</EnumVal>
    <EnumVal ord="5" desc="Function is inactive but shows its configuration capability.">off</EnumVal>
</EnumType>
<EnumType id="HealthKind" desc="Health (HealthKind enumeration)">
    <EnumVal ord="1" desc="No problems, normal operation (&quot;green&quot;).">Ok</EnumVal>
    <EnumVal ord="2" desc="Minor problems, but in safe operating mode (&quot;yellow&quot;). The exact meaning is
a local issue, depending on the dedicated function/device.">Warning</EnumVal>
    <EnumVal ord="3" desc="Severe problem, no operation possible (&quot;red&quot;).">Alarm</EnumVal>
</EnumType>
<EnumType id="CtlModelKind" desc="Control model (CtlModelKind enumeration)">
    <EnumVal ord="0" desc="The object is not controllable, only the services that apply to a status object are
supported.">status-only</EnumVal>
    <EnumVal ord="1" desc="Direct control with normal security according to IEC 61850-7-2.">direct-with-normal-
security</EnumVal>

```

```

    <EnumVal ord="2" desc="SBO (select before operate) control with normal security according to IEC 61850-7-2.">sbo-with-normal-security</EnumVal>
    <EnumVal ord="3" desc="Direct control with enhanced security according to IEC 61850-7-2.">direct-with-enhanced-security</EnumVal>
    <EnumVal ord="4" desc="SBO control with enhanced security according to IEC 61850-7-2.">sbo-with-enhanced-security</EnumVal>
  </EnumType>
  <EnumType id="FaultDirectionKind" desc="Fault direction (FaultDirectionKind enumeration)">
    <EnumVal ord="0" desc="unknown">unknown</EnumVal>
    <EnumVal ord="1" desc="forward">forward</EnumVal>
    <EnumVal ord="2" desc="backward">backward</EnumVal>
    <EnumVal ord="3" desc="both">both</EnumVal>
  </EnumType>
  <EnumType id="SequenceKind" desc="Sequence (SequenceKind enumeration)">
    <EnumVal ord="0" desc="Sequence components 'CDC.[c1, c2, c3]' are positive, negative and zero, respectively.">pos-neg-zero</EnumVal>
    <EnumVal ord="1" desc="Sequence components 'CDC.[c1, c2, c3]' are direct, quadratic and zero, respectively.">dir-quad-zero</EnumVal>
  </EnumType>
  <EnumType id="OriginatorCategoryKind" desc="Originator category (OriginatorCategoryKind enumeration)">
    <EnumVal ord="0" desc="That value shall not be used.">not-supported</EnumVal>
    <EnumVal ord="1" desc="Control operation issued from an operator using a client located at bay level.">bay-control</EnumVal>
    <EnumVal ord="2" desc="Control operation issued from an operator using a client located at station level.">station-control</EnumVal>
    <EnumVal ord="3" desc="Control operation from a remote operator outside the substation (for example network control center).">remote-control</EnumVal>
    <EnumVal ord="4" desc="Control operation issued from an automatic function at bay level.">automatic-bay</EnumVal>
    <EnumVal ord="5" desc="Control operation issued from an automatic function at station level.">automatic-station</EnumVal>
    <EnumVal ord="6" desc="Control operation issued from an automatic function outside of the substation.">automatic-remote</EnumVal>
    <EnumVal ord="7" desc="Control operation issued from a maintenance/service tool.">maintenance</EnumVal>
    <EnumVal ord="8" desc="Status change occurred without control action (for example external trip of a circuit breaker or failure inside the breaker).">process</EnumVal>
  </EnumType>
  <EnumType id="SIUnitKind" desc="SI unit (SIUnitKind enumeration)">
    <EnumVal ord="1" desc="dimensionless"></EnumVal>
    <EnumVal ord="2" desc="meter: Length">m</EnumVal>
    <EnumVal ord="3" desc="kilogram: Mass">kg</EnumVal>
    <EnumVal ord="4" desc="second: Time">s</EnumVal>
    <EnumVal ord="5" desc="ampere: Current">A</EnumVal>
    <EnumVal ord="6" desc="kelvin: Temperature">K</EnumVal>
    <EnumVal ord="7" desc="mole: Amount of substance">mol</EnumVal>
    <EnumVal ord="8" desc="candela: Luminous intensity">cd</EnumVal>
    <EnumVal ord="9" desc="degrees: Plane angle">deg</EnumVal>
    <EnumVal ord="10" desc="radian: Plane angle">rad</EnumVal>
    <EnumVal ord="11" desc="steradian: Solid angle">sr</EnumVal>
    <EnumVal ord="21" desc="Gray (J/kg): Absorbed dose">Gy</EnumVal>
    <EnumVal ord="22" desc="becquerel (1/s): Activity">Bq</EnumVal>
    <EnumVal ord="23" desc="degrees Celsius: Relative temperature">°C</EnumVal>
    <EnumVal ord="24" desc="sievert (J/kg): Dose equivalent">Sv</EnumVal>
    <EnumVal ord="25" desc="farad: Electric capacitance">F</EnumVal>
    <EnumVal ord="26" desc="coulomb (A S): Electric charge">C</EnumVal>
    <EnumVal ord="27" desc="siemens (A/V): Electric conductance">S</EnumVal>
    <EnumVal ord="28" desc="henry (Wb/A): Electric inductance">H</EnumVal>
    <EnumVal ord="29" desc="volt (W/A): Electric potential">V</EnumVal>
    <EnumVal ord="30" desc="ohm (V/A): Electric resistance">ohm</EnumVal>
    <EnumVal ord="31" desc="joule (N m): Energy">J</EnumVal>
    <EnumVal ord="32" desc="newton (kg m / s²): Force">N</EnumVal>
    <EnumVal ord="33" desc="hertz (1/s): Frequency">Hz</EnumVal>
    <EnumVal ord="34" desc="lux (Lm/m²): Illuminance">lx</EnumVal>
    <EnumVal ord="35" desc="lumen (cd sr): Luminous flux">Lm</EnumVal>
    <EnumVal ord="36" desc="weber (V s): Magnetic flux">Wb</EnumVal>
    <EnumVal ord="37" desc="tesla (Wb / m²): Magnetic flux density">T</EnumVal>
    <EnumVal ord="38" desc="watt (J/s): Powerwatt (I² R): Real power">W</EnumVal>
    <EnumVal ord="39" desc="pascal (N / m²): Pressure">Pa</EnumVal>
    <EnumVal ord="41" desc="square meter (m²): Area">m²</EnumVal>
    <EnumVal ord="42" desc="cubic meter (m³): Volume">m³</EnumVal>
    <EnumVal ord="43" desc="meters per second (m/s): Velocity">m/s</EnumVal>
    <EnumVal ord="44" desc="meters per second² (m/s²): Acceleration">m/s²</EnumVal>
    <EnumVal ord="45" desc="cubic meters per second (m³/s): Volumetric flow rate">m³/s</EnumVal>
  </EnumType>

```

```

<EnumVal ord="46" desc="meters/cubic meter (m/m³): Fuel efficiency">m/m³</EnumVal>
<EnumVal ord="47" desc="kilogram meter (kg m): Moment of mass">M</EnumVal>
<EnumVal ord="48" desc="kilogram/cubic meter (kg/m³): Density">kg/m³</EnumVal>
<EnumVal ord="49" desc="meter square/second (m²/s): Viscosity">m²/s</EnumVal>
<EnumVal ord="50" desc="watt per meter kelvin (W/m K): Thermal conductivity">W/m K</EnumVal>
<EnumVal ord="51" desc="joule per kelvin (J/K): Heat capacity">J/K</EnumVal>
<EnumVal ord="52" desc="parts per million: Concentration">ppm</EnumVal>
<EnumVal ord="53" desc="rotations per second (1/s): Rotational speed">1/s</EnumVal>
<EnumVal ord="54" desc="radian per second (rad/s): Angular velocity">rad/s</EnumVal>
<EnumVal ord="55" desc="watt per square meter: Insolation">W/m²</EnumVal>
<EnumVal ord="56" desc="watt seconds per square meter: Insolation energy">J/m²</EnumVal>
<EnumVal ord="57" desc="siemens per meter: Electric conductivity">S/m</EnumVal>
<EnumVal ord="58" desc="kelvin per second: Temperature change rate">K/s</EnumVal>
<EnumVal ord="59" desc="pascal per second: Pressure change rate">Pa/s</EnumVal>
<EnumVal ord="60" desc="joule per kilogram per kelvin: Specific heat">J/kg K</EnumVal>
<EnumVal ord="61" desc="volt ampere (VA): Apparent power">VA</EnumVal>
<EnumVal ord="62" desc="(Use 'W' instead) watts (I² R): Real power.">Watts</EnumVal>
<EnumVal ord="63" desc="volt ampere reactive (V I sin(theta)): Reactive power">VAr</EnumVal>
<EnumVal ord="64" desc="(Use 'deg' or 'rad' instead) degrees: Phase angle.">phi</EnumVal>
<EnumVal ord="65" desc="dimensionless: Power factor">cos(phi)</EnumVal>
<EnumVal ord="66" desc="volt seconds (Ws/A): Volt second">Vs</EnumVal>
<EnumVal ord="67" desc="volt squared (W²/A²): Volt squared">V²</EnumVal>
<EnumVal ord="68" desc="amp second (As): Amp second">As</EnumVal>
<EnumVal ord="69" desc="amp square (A²): Amp squared">A²</EnumVal>
<EnumVal ord="70" desc="amp squared time (A²t): Amps squared time">A²t</EnumVal>
<EnumVal ord="71" desc="volt ampere hours: Apparent energy">VAh</EnumVal>
<EnumVal ord="72" desc="watt hours: Real energy">Wh</EnumVal>
<EnumVal ord="73" desc="volt ampere reactive hours: Reactive energy">VArh</EnumVal>
<EnumVal ord="74" desc="volts per hertz: Magnetic flux">V/Hz</EnumVal>
<EnumVal ord="75" desc="hertz per second: Rate of change of frequency">Hz/s</EnumVal>
<EnumVal ord="76" desc="characters: Number of characters">char</EnumVal>
<EnumVal ord="77" desc="characters per second: Baud">char/s</EnumVal>
<EnumVal ord="78" desc="kg square meter: Turbine inertia">kgm²</EnumVal>
<EnumVal ord="79" desc="decibel: Sound pressure level">dB</EnumVal>
<EnumVal ord="80" desc="joule per watt-hour: Heat rate">J/Wh</EnumVal>
<EnumVal ord="81" desc="watt per second: Ramp rate">W/s</EnumVal>
<EnumVal ord="82" desc="litres per second: Flow rate">l/s</EnumVal>
<EnumVal ord="83" desc="power measurement relative to 1 mW: Power level">dBm</EnumVal>
<EnumVal ord="84" desc="hour: Time">h</EnumVal>
<EnumVal ord="85" desc="minute: Time">min</EnumVal>
</EnumType>
<EnumType id="MultiplierKind" desc="Unit multiplier (MultiplierKind enumeration)">
  <EnumVal ord="-24" desc="yocto (10exp(-24))">y</EnumVal>
  <EnumVal ord="-21" desc="zepto (10exp(-21))">z</EnumVal>
  <EnumVal ord="-18" desc="atto (10exp(-18))">a</EnumVal>
  <EnumVal ord="-15" desc="femto (10exp(-15))">f</EnumVal>
  <EnumVal ord="-12" desc="pico (10exp(-12))">p</EnumVal>
  <EnumVal ord="-9" desc="nano (10exp(-9))">n</EnumVal>
  <EnumVal ord="-6" desc="micro (10exp(-6))">µ</EnumVal>
  <EnumVal ord="-3" desc="milli (10exp(-3))">m</EnumVal>
  <EnumVal ord="-2" desc="centi (10exp(-2))">c</EnumVal>
  <EnumVal ord="-1" desc="deci (10exp(-1))">d</EnumVal>
  <EnumVal ord="0" desc="1 (10exp(0))"/>
  <EnumVal ord="1" desc="deca (10exp(1))">da</EnumVal>
  <EnumVal ord="2" desc="hecto (10exp(2))">h</EnumVal>
  <EnumVal ord="3" desc="kilo (10exp(3))">k</EnumVal>
  <EnumVal ord="6" desc="mega (10exp(6))">M</EnumVal>
  <EnumVal ord="9" desc="giga (10exp(9))">G</EnumVal>
  <EnumVal ord="12" desc="tera (10exp(12))">T</EnumVal>
  <EnumVal ord="15" desc="petra (10exp(15))">P</EnumVal>
  <EnumVal ord="18" desc="exa (10exp(18))">E</EnumVal>
  <EnumVal ord="21" desc="zetta (10exp(21))">Z</EnumVal>
  <EnumVal ord="24" desc="yotta (10exp(24))">Y</EnumVal>
</EnumType>
</DataTypeTemplates>
</SCL>

```

## Annex E (informative)

### E.2 Base types

Replace the existing text of Subclause E.2 with the following new text:

#### SCL\_BaseSimpleTypes.xsd

##### <CODE BEGINS>

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns="http://www.iec.ch/61850/2003/SCL"
targetNamespace="http://www.iec.ch/61850/2003/SCL" elementFormDefault="qualified" attributeFormDefault="unqualified"
version="Mixed.2007B4">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Informative general SCL schema for mixed systems version "2007" revision "B" release 4, for
      IEC 61850-6 Ed. 2.1.
```

COPYRIGHT (c) IEC, 2016. All rights reserved. Disclaimer: The IEC disclaims liability for any personal injury, property or other damages of any nature whatsoever, whether special, indirect, consequential or compensatory, directly or indirectly resulting from this software and the document upon which its methods are based, use of, or reliance upon.

```
    </xs:documentation>
  </xs:annotation>
  <xs:simpleType name="tConnectivityNodeReference">
    <xs:restriction base="xs:normalizedString">
      <xs:pattern value="\.+\.+(\/.+)*"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tAnyName">
    <xs:restriction base="xs:normalizedString"/>
  </xs:simpleType>
  <xs:simpleType name="tName">
    <xs:restriction base="tAnyName">
      <xs:minLength value="1"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tID">
    <xs:restriction base="xs:token">
      <xs:minLength value="1"/>
      <xs:maxLength value="255"/>
      <xs:pattern value="\S+"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tAccsiName">
    <xs:restriction base="xs:Name">
      <xs:pattern value="[A-Za-z][0-9A-Za-z_]*"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tRestrName1stU">
    <xs:restriction base="xs:Name">
      <xs:pattern value="[A-Z][0-9A-Za-z]*"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tRestrName1stL">
    <xs:restriction base="xs:Name">
      <xs:pattern value="[a-z][0-9A-Za-z]*"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tPAddr">
    <xs:restriction base="xs:normalizedString">
      <xs:minLength value="1"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tSclVersion">
    <xs:restriction base="tName">
```

```
<xs:pattern value="2[0-2][0-9]{2}"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="tSciRevision">
  <xs:restriction base="xs:Name">
    <xs:pattern value="[A-Z]"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tSciRelease">
  <xs:restriction base="xs:unsignedByte">
    <xs:minExclusive value="0"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tEmpty">
  <xs:restriction base="xs:normalizedString">
    <xs:maxLength value="0"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tIEDName">
  <xs:restriction base="tAcsiName">
    <xs:maxLength value="64"/>
    <xs:pattern value="[A-Za-z][0-9A-Za-z_]{0,2}"/>
    <xs:pattern value="[A-Za-z][0-9A-Za-z_]{4,63}"/>
    <xs:pattern value="[A-MO-Za-z][0-9A-Za-z_]{3}"/>
    <xs:pattern value="N[0-9A-Za-z-mp-z_][0-9A-Za-z_]{2}"/>
    <xs:pattern value="No[0-9A-Za-z-mo-z_][0-9A-Za-z_]/>
    <xs:pattern value="Non[0-9A-Za-z-mp-z_]/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tIEDNameIsNone">
  <xs:restriction base="tAcsiName">
    <xs:pattern value="None"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tIEDNameOrNone">
  <xs:union memberTypes="tIEDName tIEDNameIsNone"/>
</xs:simpleType>
<xs:simpleType name="tOnlyRelativeIEDName">
  <xs:restriction base="xs:normalizedString">
    <xs:pattern value="&#x0040;"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tIEDNameOrRelative">
  <xs:union memberTypes="tIEDName tOnlyRelativeIEDName"/>
</xs:simpleType>
<xs:simpleType name="tLDName">
  <xs:restriction base="xs:normalizedString">
    <xs:maxLength value="64"/>
    <xs:pattern value="[A-Za-z][0-9A-Za-z_]*"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tLDInst">
  <xs:restriction base="xs:normalizedString">
    <xs:maxLength value="64"/>
    <xs:pattern value="[A-Za-z0-9][0-9A-Za-z_]*"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tLDInstOrEmpty">
  <xs:union memberTypes="tLDInst tEmpty"/>
</xs:simpleType>
<xs:simpleType name="tPrefix">
  <xs:restriction base="xs:normalizedString">
    <xs:maxLength value="11"/>
    <xs:pattern value="[A-Za-z][0-9A-Za-z_]*"/>
    <xs:pattern value=""/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tLNInst">
  <xs:restriction base="xs:normalizedString">
    <xs:pattern value="[0-9]{1,12}"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tLNInstOrEmpty">
```



```

    <xs:union memberTypes="tLNInst tEmpty"/>
  </xs:simpleType>
  <xs:simpleType name="tDataName">
    <xs:restriction base="tRestrName1stU">
      <xs:maxLength value="12"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tDataSetName">
    <xs:restriction base="tAcsiName">
      <xs:maxLength value="32"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tCBName">
    <xs:restriction base="tAcsiName">
      <xs:maxLength value="32"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tLogName">
    <xs:restriction base="tAcsiName">
      <xs:maxLength value="32"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tAccessPointName">
    <xs:restriction base="xs:normalizedString">
      <xs:pattern value="[A-Za-z0-9][0-9A-Za-z_]*"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tAssociationID">
    <xs:restriction base="xs:normalizedString">
      <xs:minLength value="1"/>
      <xs:pattern value="[0-9A-Za-z]"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tVisibleBasicLatin">
    <xs:restriction base="xs:normalizedString">
      <xs:pattern value="[#x0020;-&#x007E;]*"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tMessageID">
    <xs:restriction base="tVisibleBasicLatin">
      <xs:minLength value="1"/>
      <xs:maxLength value="129"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tFullAttributeName">
    <xs:restriction base="xs:normalizedString">
      <xs:pattern value="[a-zA-Z][a-zA-Z0-9]*(\{[0-9]+\})?(\.[a-zA-Z][a-zA-Z0-9]*(\{[0-9]+\})?)*/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tFullIDOName">
    <xs:restriction base="xs:normalizedString">
      <xs:pattern value="[A-Z][0-9A-Za-z]{0,11}(\.[a-z][0-9A-Za-z]*(\{[0-9]+\})?)?"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tSubDataName">
    <xs:restriction base="tRestrName1stL">
      <xs:minLength value="1"/>
      <xs:maxLength value="60"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tNamespaceName">
    <xs:restriction base="xs:normalizedString">
      <xs:pattern value="[&#x0020;-&#x007E;]+:20\d[d[A-Z]]?"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tLineType">
    <xs:restriction base="xs:normalizedString">
      <xs:minLength value="1"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tProcessType">
    <xs:restriction base="xs:normalizedString">
      <xs:minLength value="1"/>
    </xs:restriction>
  </xs:simpleType>

```

```

        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="tProcessName">
        <xs:restriction base="xs:normalizedString">
            <xs:pattern value=".(/.+)*"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="tEnumStringValue">
        <xs:restriction base="xs:normalizedString">
            <xs:maxLength value="127"/>
            <xs:pattern value="[\p{IsBasicLatin}\p{IsLatin-1Supplement}]*"/>
        </xs:restriction>
    </xs:simpleType>
</xs:schema>

```

<CODE ENDS>

**SCL\_Enums.xsd**

<CODE BEGINS>

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:scl="http://www.iec.ch/61850/2003/SCL" xmlns="http://www.iec.ch/61850/2003/SCL"
xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="http://www.iec.ch/61850/2003/SCL"
elementFormDefault="qualified" attributeFormDefault="unqualified" version="Mixed.2007B4">

```

```

    <xs:annotation>
        <xs:documentation xml:lang="en">
            Informative general SCL schema for mixed systems version "2007" revision "B" release 4, for
            IEC 61850-6 Ed. 2.1.

```

COPYRIGHT (c) IEC, 2016. All rights reserved. Disclaimer: The IEC disclaims liability for any personal injury, property or other damages of any nature whatsoever, whether special, indirect, consequential or compensatory, directly or indirectly resulting from this software and the document upon which its methods are based, use of, or reliance upon.

```

        </xs:documentation>
    </xs:annotation>
    <xs:include schemaLocation="SCL_BaseSimpleTypes.xsd"/>
    <xs:simpleType name="tPredefinedPTypeEnum">
        <xs:restriction base="xs:Name">
            <xs:enumeration value="IP"/>
            <xs:enumeration value="IP-SUBNET"/>
            <xs:enumeration value="IP-GATEWAY"/>
            <xs:enumeration value="OSI-NSAP"/>
            <xs:enumeration value="OSI-TSEL"/>
            <xs:enumeration value="OSI-SSEL"/>
            <xs:enumeration value="OSI-PSEL"/>
            <xs:enumeration value="OSI-AP-Title"/>
            <xs:enumeration value="OSI-AP-Invoke"/>
            <xs:enumeration value="OSI-AE-Qualifier"/>
            <xs:enumeration value="OSI-AE-Invoke"/>
            <xs:enumeration value="MAC-Address"/>
            <xs:enumeration value="APPID"/>
            <xs:enumeration value="VLAN-PRIORITY"/>
            <xs:enumeration value="VLAN-ID"/>
            <xs:enumeration value="SNTP-Port"/>
            <xs:enumeration value="MMS-Port"/>
            <xs:enumeration value="DNSName"/>
            <xs:enumeration value="IPv6FlowLabel"/>
            <xs:enumeration value="IPv6ClassOfTraffic"/>
            <xs:enumeration value="C37-118-IP-Port"/>
            <xs:enumeration value="IP-UDP-PORT"/>
            <xs:enumeration value="IP-TCP-PORT"/>
            <xs:enumeration value="IPv6"/>
            <xs:enumeration value="IPv6-SUBNET"/>
            <xs:enumeration value="IPv6-GATEWAY"/>
            <xs:enumeration value="IPv6-IGMPv3Src"/>
            <xs:enumeration value="IP-IGMPv3Src"/>
            <xs:enumeration value="IP-ClassOfTraffic"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="tExtensionPTypeEnum">
        <xs:restriction base="xs:normalizedString">
            <xs:pattern value="[A-Z][0-9A-Za-z\-\_]*"/>

```

```
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="tPTypeEnum">
  <xs:union memberTypes="tPredefinedPTypeEnum tExtensionPTypeEnum"/>
</xs:simpleType>
<xs:simpleType name="tPredefinedPTypeEnumPhysConnEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="Type"/>
    <xs:enumeration value="Plug"/>
    <xs:enumeration value="Cable"/>
    <xs:enumeration value="Port"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tPTypeEnumPhysConnEnum">
  <xs:union memberTypes="tPredefinedPTypeEnumPhysConnEnum tExtensionPTypeEnum"/>
</xs:simpleType>
<xs:simpleType name="tPredefinedAttributeNameEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="T"/>
    <xs:enumeration value="Test"/>
    <xs:enumeration value="Check"/>
    <xs:enumeration value="SIUnit"/>
    <xs:enumeration value="Oper"/>
    <xs:enumeration value="SBO"/>
    <xs:enumeration value="SBOw"/>
    <xs:enumeration value="Cancel"/>
    <xs:enumeration value="Addr"/>
    <xs:enumeration value="PRIORITY"/>
    <xs:enumeration value="VID"/>
    <xs:enumeration value="APPID"/>
    <xs:enumeration value="TransportInUse"/>
    <xs:enumeration value="IPClassOfTraffic"/>
    <xs:enumeration value="IPv6FlowLabel"/>
    <xs:enumeration value="IPAddressLength"/>
    <xs:enumeration value="IPAddress"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tExtensionAttributeNameEnum">
  <xs:restriction base="tRestrName1stL">
    <xs:maxLength value="60"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tAttributeNameEnum">
  <xs:union memberTypes="tPredefinedAttributeNameEnum tExtensionAttributeNameEnum"/>
</xs:simpleType>
<xs:simpleType name="tPredefinedCommonConductingEquipmentEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="CBR"/>
    <xs:enumeration value="DIS"/>
    <xs:enumeration value="VTR"/>
    <xs:enumeration value="CTR"/>
    <xs:enumeration value="GEN"/>
    <xs:enumeration value="CAP"/>
    <xs:enumeration value="REA"/>
    <xs:enumeration value="CON"/>
    <xs:enumeration value="MOT"/>
    <xs:enumeration value="EFN"/>
    <xs:enumeration value="PSH"/>
    <xs:enumeration value="BAT"/>
    <xs:enumeration value="BSH"/>
    <xs:enumeration value="CAB"/>
    <xs:enumeration value="GIL"/>
    <xs:enumeration value="LIN"/>
    <xs:enumeration value="RES"/>
    <xs:enumeration value="RRC"/>
    <xs:enumeration value="SAR"/>
    <xs:enumeration value="TCF"/>
    <xs:enumeration value="TCR"/>
    <xs:enumeration value="IFL"/>
    <xs:enumeration value="FAN"/>
    <xs:enumeration value="SCR"/>
    <xs:enumeration value="SMC"/>
    <xs:enumeration value="PMP"/>
  </xs:restriction>
</xs:simpleType>
```

IECNORM.COM - IEC 61850-6:2009/AMD1:2018

```

    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tExtensionEquipmentEnum">
  <xs:restriction base="xs:Name">
    <xs:minLength value="3"/>
    <xs:pattern value="E[A-Z]*"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tCommonConductingEquipmentEnum">
  <xs:union memberTypes="tPredefinedCommonConductingEquipmentEnum tExtensionEquipmentEnum"/>
</xs:simpleType>
<xs:simpleType name="tPowerTransformerEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="PTR"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tTransformerWindingEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="PTW"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tPredefinedGeneralEquipmentEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="AXN"/>
    <xs:enumeration value="BAT"/>
    <xs:enumeration value="MOT"/>
    <xs:enumeration value="FAN"/>
    <xs:enumeration value="FIL"/>
    <xs:enumeration value="PMP"/>
    <xs:enumeration value="TNK"/>
    <xs:enumeration value="VLV"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tExtensionGeneralEquipmentEnum">
  <xs:restriction base="xs:Name">
    <xs:minLength value="3"/>
    <xs:pattern value="E[A-Z]*"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tGeneralEquipmentEnum">
  <xs:union memberTypes="tPredefinedGeneralEquipmentEnum tExtensionGeneralEquipmentEnum"/>
</xs:simpleType>
<xs:simpleType name="tServiceSettingsNoDynEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="Conf"/>
    <xs:enumeration value="Fix"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tServiceSettingsEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="Dyn"/>
    <xs:enumeration value="Conf"/>
    <xs:enumeration value="Fix"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tRedProtEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="none"/>
    <xs:enumeration value="hsr"/>
    <xs:enumeration value="prp"/>
    <xs:enumeration value="rstp"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tSMVDeliveryEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="unicast"/>
    <xs:enumeration value="multicast"/>
    <xs:enumeration value="both"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tPhaseEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="A"/>

```



```
<xs:enumeration value="B"/>
<xs:enumeration value="C"/>
<xs:enumeration value="N"/>
<xs:enumeration value="all"/>
<xs:enumeration value="none"/>
<xs:enumeration value="AB"/>
<xs:enumeration value="BC"/>
<xs:enumeration value="CA"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="tAuthenticationEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="none"/>
    <xs:enumeration value="password"/>
    <xs:enumeration value="weak"/>
    <xs:enumeration value="strong"/>
    <xs:enumeration value="certificate"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tAssociationKindEnum">
  <xs:restriction base="xs:token">
    <xs:enumeration value="pre-established"/>
    <xs:enumeration value="predefined"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tLPHDEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="LPHD"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tLLN0Enum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="LLN0"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tSystemLNGroupEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="L[A-Z]*"/>
    <xs:pattern value="LLN0"/>
    <xs:enumeration value="LLN0"/>
    <xs:enumeration value="LPHD"/>
    <xs:enumeration value="LCCH"/>
    <xs:enumeration value="LGOS"/>
    <xs:enumeration value="LSVS"/>
    <xs:enumeration value="LTIM"/>
    <xs:enumeration value="LTMS"/>
    <xs:enumeration value="LTRK"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupAEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="A[A-Z]*"/>
    <xs:enumeration value="ANCR"/>
    <xs:enumeration value="ARCO"/>
    <xs:enumeration value="ARIS"/>
    <xs:enumeration value="ATCC"/>
    <xs:enumeration value="AVCO"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupCEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="C[A-Z]*"/>
    <xs:enumeration value="CALH"/>
    <xs:enumeration value="CCGR"/>
    <xs:enumeration value="CILO"/>
    <xs:enumeration value="CPOW"/>
    <xs:enumeration value="CSWI"/>
    <xs:enumeration value="CSYN"/>
  </xs:restriction>
</xs:simpleType>
```

IEC NORMATIVE DOCUMENT  
View the full PDF of IEC 61850-6:2009/AMD1:2018

```

<xs:simpleType name="tDomainLNGroupFEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="F[A-Z]*/>
    <xs:enumeration value="FCNT"/>
    <xs:enumeration value="FCSD"/>
    <xs:enumeration value="FFIL"/>
    <xs:enumeration value="FLIM"/>
    <xs:enumeration value="FPID"/>
    <xs:enumeration value="FRMP"/>
    <xs:enumeration value="FSPT"/>
    <xs:enumeration value="FXOT"/>
    <xs:enumeration value="FXUT"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupGEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="G[A-Z]*/>
    <xs:enumeration value="GAPC"/>
    <xs:enumeration value="GGIO"/>
    <xs:enumeration value="GLOG"/>
    <xs:enumeration value="GSAL"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupIEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="I[A-Z]*/>
    <xs:enumeration value="IARC"/>
    <xs:enumeration value="IHMI"/>
    <xs:enumeration value="ISAF"/>
    <xs:enumeration value="ITCI"/>
    <xs:enumeration value="ITMI"/>
    <xs:enumeration value="ITPC"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupKEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="K[A-Z]*/>
    <xs:enumeration value="KFAN"/>
    <xs:enumeration value="KFIL"/>
    <xs:enumeration value="KPMP"/>
    <xs:enumeration value="KTNK"/>
    <xs:enumeration value="KVLV"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupMEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="M[A-Z]*/>
    <xs:enumeration value="MDIF"/>
    <xs:enumeration value="MENV"/>
    <xs:enumeration value="MFLK"/>
    <xs:enumeration value="MHAI"/>
    <xs:enumeration value="MHAN"/>
    <xs:enumeration value="MHYD"/>
    <xs:enumeration value="MMDC"/>
    <xs:enumeration value="MMET"/>
    <xs:enumeration value="MMTN"/>
    <xs:enumeration value="MMTR"/>
    <xs:enumeration value="MMXN"/>
    <xs:enumeration value="MMXU"/>
    <xs:enumeration value="MSQI"/>
    <xs:enumeration value="MSTA"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupPEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="P[A-Z]*/>
    <xs:enumeration value="PDIF"/>
  </xs:restriction>

```

IECNORM.COM: Copyright © IEC 2018. To view the full PDF of IEC 61850-6:2009/AMD1:2018

```
<xs:enumeration value="PDIR"/>
<xs:enumeration value="PDIS"/>
<xs:enumeration value="PDOP"/>
<xs:enumeration value="PDUP"/>
<xs:enumeration value="PFRG"/>
<xs:enumeration value="PHAR"/>
<xs:enumeration value="PHIZ"/>
<xs:enumeration value="PIOC"/>
<xs:enumeration value="PMRI"/>
<xs:enumeration value="PMSS"/>
<xs:enumeration value="POPF"/>
<xs:enumeration value="PPAM"/>
<xs:enumeration value="PRTR"/>
<xs:enumeration value="PSCH"/>
<xs:enumeration value="PSDE"/>
<xs:enumeration value="PTEF"/>
<xs:enumeration value="PTHF"/>
<xs:enumeration value="PTOC"/>
<xs:enumeration value="PTOF"/>
<xs:enumeration value="PTOV"/>
<xs:enumeration value="PTRC"/>
<xs:enumeration value="PTTR"/>
<xs:enumeration value="PTUC"/>
<xs:enumeration value="PTUF"/>
<xs:enumeration value="PTUV"/>
<xs:enumeration value="PUPF"/>
<xs:enumeration value="PVOC"/>
<xs:enumeration value="PVPH"/>
<xs:enumeration value="PZSU"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupQEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="Q[A-Z]*"/>
    <xs:enumeration value="QFVR"/>
    <xs:enumeration value="QITR"/>
    <xs:enumeration value="QIUB"/>
    <xs:enumeration value="QVTR"/>
    <xs:enumeration value="QVUB"/>
    <xs:enumeration value="QVVR"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupREnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="R[A-Z]*"/>
    <xs:enumeration value="RADR"/>
    <xs:enumeration value="RBDR"/>
    <xs:enumeration value="RBRF"/>
    <xs:enumeration value="RDIR"/>
    <xs:enumeration value="RDRE"/>
    <xs:enumeration value="RDRS"/>
    <xs:enumeration value="RFLO"/>
    <xs:enumeration value="RMXU"/>
    <xs:enumeration value="RPSB"/>
    <xs:enumeration value="RREC"/>
    <xs:enumeration value="RSYN"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupSEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="S[A-Z]*"/>
    <xs:enumeration value="SARC"/>
    <xs:enumeration value="SCBR"/>
    <xs:enumeration value="SIMG"/>
    <xs:enumeration value="SIML"/>
    <xs:enumeration value="SLTC"/>
    <xs:enumeration value="SOPM"/>
    <xs:enumeration value="SPDC"/>
    <xs:enumeration value="SPTR"/>
    <xs:enumeration value="SSWI"/>
  </xs:restriction>
</xs:simpleType>
```

IECNORM.COM - Full PDF of IEC 61850-6:2009/AMD1:2018

```
<xs:enumeration value="STMP"/>
<xs:enumeration value="SVBR"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupTEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="T[A-Z]*"/>
    <xs:enumeration value="TANG"/>
    <xs:enumeration value="TAXD"/>
    <xs:enumeration value="TCTR"/>
    <xs:enumeration value="TDST"/>
    <xs:enumeration value="TFLW"/>
    <xs:enumeration value="TFRQ"/>
    <xs:enumeration value="TGSN"/>
    <xs:enumeration value="THUM"/>
    <xs:enumeration value="TLVL"/>
    <xs:enumeration value="TMGF"/>
    <xs:enumeration value="TMVM"/>
    <xs:enumeration value="TPOS"/>
    <xs:enumeration value="TPRS"/>
    <xs:enumeration value="TRTN"/>
    <xs:enumeration value="TSND"/>
    <xs:enumeration value="TTMP"/>
    <xs:enumeration value="TTNS"/>
    <xs:enumeration value="TVBR"/>
    <xs:enumeration value="TVTR"/>
    <xs:enumeration value="TWPH"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupXEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="X[A-Z]*"/>
    <xs:enumeration value="XCBR"/>
    <xs:enumeration value="XSWI"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupYEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="Y[A-Z]*"/>
    <xs:enumeration value="YEFN"/>
    <xs:enumeration value="YLTC"/>
    <xs:enumeration value="YPSH"/>
    <xs:enumeration value="YPTR"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNGroupZEnum">
  <xs:restriction base="xs:Name">
    <xs:length value="4"/>
    <xs:pattern value="Z[A-Z]*"/>
    <xs:enumeration value="ZAXN"/>
    <xs:enumeration value="ZBAT"/>
    <xs:enumeration value="ZBSH"/>
    <xs:enumeration value="ZCAB"/>
    <xs:enumeration value="ZCAP"/>
    <xs:enumeration value="ZCON"/>
    <xs:enumeration value="ZGEN"/>
    <xs:enumeration value="ZGIL"/>
    <xs:enumeration value="ZLIN"/>
    <xs:enumeration value="ZMOT"/>
    <xs:enumeration value="ZREA"/>
    <xs:enumeration value="ZRES"/>
    <xs:enumeration value="ZRRC"/>
    <xs:enumeration value="ZSAR"/>
    <xs:enumeration value="ZSCR"/>
    <xs:enumeration value="ZSMC"/>
    <xs:enumeration value="ZTCF"/>
    <xs:enumeration value="ZTCR"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tDomainLNEnum">
```



```

    <xs:union memberTypes="tDomainLNGroupAEnum tDomainLNGroupCEnum tDomainLNGroupFEnum
tDomainLNGroupGEnum tDomainLNGroupIEnum tDomainLNGroupKEnum tDomainLNGroupMEnum tDomainLNGroupPEnum
tDomainLNGroupQEnum tDomainLNGroupREnum tDomainLNGroupSEnum tDomainLNGroupTEnum tDomainLNGroupXEnum
tDomainLNGroupYEnum tDomainLNGroupZEnum"/>

```

```

</xs:simpleType>

```

```

<xs:simpleType name="tPredefinedLNClassEnum">

```

```

  <xs:union memberTypes="tSystemLNGroupEnum tDomainLNEnum"/>

```

```

</xs:simpleType>

```

```

<xs:simpleType name="tExtensionLNClassEnum">

```

```

  <xs:restriction base="xs:Name">

```

```

    <xs:length value="4"/>

```

```

    <xs:pattern value="[A-Z]+"/>

```

```

  </xs:restriction>

```

```

</xs:simpleType>

```

```

<xs:simpleType name="tLNClassEnum">

```

```

  <xs:union memberTypes="tPredefinedLNClassEnum tExtensionLNClassEnum"/>

```

```

</xs:simpleType>

```

```

<xs:simpleType name="tPredefinedCDCEnum">

```

```

  <xs:restriction base="xs:Name">

```

```

    <xs:enumeration value="SPS"/>

```

```

    <xs:enumeration value="DPS"/>

```

```

    <xs:enumeration value="INS"/>

```

```

    <xs:enumeration value="ENS"/>

```

```

    <xs:enumeration value="ACT"/>

```

```

    <xs:enumeration value="ACD"/>

```

```

    <xs:enumeration value="SEC"/>

```

```

    <xs:enumeration value="BCR"/>

```

```

    <xs:enumeration value="HST"/>

```

```

    <xs:enumeration value="VSS"/>

```

```

    <xs:enumeration value="MV"/>

```

```

    <xs:enumeration value="CMV"/>

```

```

    <xs:enumeration value="SAV"/>

```

```

    <xs:enumeration value="WYE"/>

```

```

    <xs:enumeration value="DEL"/>

```

```

    <xs:enumeration value="SEQ"/>

```

```

    <xs:enumeration value="HMV"/>

```

```

    <xs:enumeration value="HWYE"/>

```

```

    <xs:enumeration value="HDEL"/>

```

```

    <xs:enumeration value="SPC"/>

```

```

    <xs:enumeration value="DPC"/>

```

```

    <xs:enumeration value="INC"/>

```

```

    <xs:enumeration value="ENC"/>

```

```

    <xs:enumeration value="BSC"/>

```

```

    <xs:enumeration value="ISC"/>

```

```

    <xs:enumeration value="APC"/>

```

```

    <xs:enumeration value="BAC"/>

```

```

    <xs:enumeration value="SPG"/>

```

```

    <xs:enumeration value="ING"/>

```

```

    <xs:enumeration value="ENG"/>

```

```

    <xs:enumeration value="ORG"/>

```

```

    <xs:enumeration value="TSG"/>

```

```

    <xs:enumeration value="CUG"/>

```

```

    <xs:enumeration value="VSG"/>

```

```

    <xs:enumeration value="ASG"/>

```

```

    <xs:enumeration value="CURVE"/>

```

```

    <xs:enumeration value="CSG"/>

```

```

    <xs:enumeration value="DPL"/>

```

```

    <xs:enumeration value="LPL"/>

```

```

    <xs:enumeration value="CSD"/>

```

```

    <xs:enumeration value="CST"/>

```

```

    <xs:enumeration value="BTS"/>

```

```

    <xs:enumeration value="UTS"/>

```

```

    <xs:enumeration value="LTS"/>

```

```

    <xs:enumeration value="GTS"/>

```

```

    <xs:enumeration value="MTS"/>

```

```

    <xs:enumeration value="NTS"/>

```

```

    <xs:enumeration value="STS"/>

```

```

    <xs:enumeration value="CTS"/>

```

```

    <xs:enumeration value="OTS"/>

```

```

    <xs:enumeration value="VSD"/>

```

```

    <xs:enumeration value="ORS"/>

```

```

    <xs:enumeration value="TCS"/>

```

```

  </xs:restriction>

```

```

</xs:simpleType>

```

IECNORM.COM Full PDF of IEC 61850-6:2009/AMD1:2018

```
</xs:simpleType>
<xs:simpleType name="tExtensionCDCEnum">
  <xs:restriction base="xs:Name">
    <xs:minLength value="1"/>
    <xs:maxLength value="5"/>
    <xs:pattern value="[A-Za-z]+"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tCDCEnum">
  <xs:union memberTypes="tPredefinedCDCEnum tExtensionCDCEnum"/>
</xs:simpleType>
<xs:simpleType name="tFCEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="ST"/>
    <xs:enumeration value="MX"/>
    <xs:enumeration value="CO"/>
    <xs:enumeration value="SP"/>
    <xs:enumeration value="SG"/>
    <xs:enumeration value="SE"/>
    <xs:enumeration value="SV"/>
    <xs:enumeration value="CF"/>
    <xs:enumeration value="DC"/>
    <xs:enumeration value="EX"/>
    <xs:enumeration value="SR"/>
    <xs:enumeration value="BL"/>
    <xs:enumeration value="OR"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tPredefinedBasicTypeEnum">
  <xs:restriction base="xs:Name">
    <xs:enumeration value="BOOLEAN"/>
    <xs:enumeration value="INT8"/>
    <xs:enumeration value="INT16"/>
    <xs:enumeration value="INT24"/>
    <xs:enumeration value="INT32"/>
    <xs:enumeration value="INT64"/>
    <xs:enumeration value="INT128"/>
    <xs:enumeration value="INT8U"/>
    <xs:enumeration value="INT16U"/>
    <xs:enumeration value="INT24U"/>
    <xs:enumeration value="INT32U"/>
    <xs:enumeration value="FLOAT32"/>
    <xs:enumeration value="FLOAT64"/>
    <xs:enumeration value="Enum"/>
    <xs:enumeration value="Dbpos"/>
    <xs:enumeration value="Tcmd"/>
    <xs:enumeration value="Quality"/>
    <xs:enumeration value="Timestamp"/>
    <xs:enumeration value="VisString32"/>
    <xs:enumeration value="VisString64"/>
    <xs:enumeration value="VisString65"/>
    <xs:enumeration value="VisString129"/>
    <xs:enumeration value="VisString255"/>
    <xs:enumeration value="Octet64"/>
    <xs:enumeration value="Unicode255"/>
    <xs:enumeration value="Struct"/>
    <xs:enumeration value="EntryTime"/>
    <xs:enumeration value="Check"/>
    <xs:enumeration value="ObjRef"/>
    <xs:enumeration value="Currency"/>
    <xs:enumeration value="PhyComAddr"/>
    <xs:enumeration value="TrgOps"/>
    <xs:enumeration value="OptFlds"/>
    <xs:enumeration value="SvOptFlds"/>
    <xs:enumeration value="LogOptFlds"/>
    <xs:enumeration value="EntryID"/>
    <xs:enumeration value="Octet6"/>
    <xs:enumeration value="Octet16"/>
    <!-- for 61850-8-1 Edition 2.1 -->
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tExtensionBasicTypeEnum">
  <xs:restriction base="xs:Name">
```





```

        <xs:pattern value="[A-Z][0-9A-Za-z\-\]*"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tPhysConnTypeEnum">
    <xs:union memberTypes="tPredefinedPhysConnTypeEnum tExtensionPhysConnTypeEnum"/>
</xs:simpleType>
<xs:simpleType name="tServiceType">
    <xs:restriction base="xs:Name">
        <xs:enumeration value="Poll"/>
        <xs:enumeration value="Report"/>
        <xs:enumeration value="GOOSE"/>
        <xs:enumeration value="SMV"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="tPredefinedTypeOfSecurityEnum">
    <xs:restriction base="xs:normalizedString">
        <xs:enumeration value="None"/>
        <xs:enumeration value="Signature"/>
        <xs:enumeration value="SignatureAndEncryption"/>
    </xs:restriction>
</xs:simpleType>
</xs:schema>

```

<CODE ENDS>

## SCL\_BaseTypes.xsd

<CODE BEGINS>

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:scl="http://www.iec.ch/61850/2003/SCL" xmlns="http://www.iec.ch/61850/2003/SCL"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="http://www.iec.ch/61850/2003/SCL"
  elementFormDefault="qualified" attributeFormDefault="unqualified" version="Mixed.2007B4">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Informative general SCL schema for mixed systems version "2007" revision "B" release 4, for IEC 61850-6 Ed. 2.1.
        </xs:documentation>
        <xs:documentation>
            COPYRIGHT (c) IEC, 2016. All rights reserved. Disclaimer: The IEC disclaims liability for any personal injury,
            property or other damages of any nature whatsoever, whether special, indirect, consequential or compensatory, directly or
            indirectly resulting from this software and the document upon which its methods are based, use of, or reliance upon.
        </xs:documentation>
    </xs:annotation>
    <xs:include schemaLocation="SCL_Enums.xsd"/>
    <xs:attributeGroup name="agDesc">
        <xs:attribute name="desc" type="xs:normalizedString" use="optional" default=""/>
    </xs:attributeGroup>
    <xs:complexType name="tBaseElement" abstract="true">
        <xs:sequence>
            <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element name="Text" type="tText" minOccurs="0"/>
            <xs:element name="Private" type="tPrivate" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:anyAttribute namespace="##other" processContents="lax"/>
    </xs:complexType>
    <xs:complexType name="tUnNaming" abstract="true">
        <xs:complexContent>
            <xs:extension base="tBaseElement">
                <xs:attributeGroup ref="agDesc"/>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="tNaming" abstract="true">
        <xs:complexContent>
            <xs:extension base="tBaseElement">
                <xs:attribute name="name" type="tName" use="required"/>
                <xs:attributeGroup ref="agDesc"/>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="tIDNaming" abstract="true">
        <xs:complexContent>

```

```
<xs:extension base="tBaseElement">
  <xs:attribute name="id" type="tID" use="required"/>
  <xs:attributeGroup ref="agDesc"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tAnyContentFromOtherNamespace" abstract="true" mixed="true">
  <xs:sequence minOccurs="0" maxOccurs="unbounded">
    <xs:any namespace="##other" processContents="lax"/>
  </xs:sequence>
  <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>
<xs:complexType name="tText" mixed="true">
  <xs:complexContent>
    <xs:extension base="tAnyContentFromOtherNamespace">
      <xs:attribute name="source" type="xs:anyURI" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tPrivate" mixed="true">
  <xs:complexContent>
    <xs:extension base="tAnyContentFromOtherNamespace">
      <xs:attribute name="type" use="required">
        <xs:simpleType>
          <xs:restriction base="xs:normalizedString">
            <xs:minLength value="1"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
      <xs:attribute name="source" type="xs:anyURI" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tHeader">
  <xs:sequence>
    <xs:element name="Text" type="tText" minOccurs="0"/>
    <xs:element name="History" minOccurs="0">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="Hitem" type="tHitem" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
  <xs:attribute name="id" type="xs:normalizedString" use="required"/>
  <xs:attribute name="version" type="xs:normalizedString" use="optional"/>
  <xs:attribute name="revision" type="xs:normalizedString" use="optional" default=""/>
  <xs:attribute name="toolID" type="xs:normalizedString" use="optional"/>
  <xs:attribute name="nameStructure" use="optional" default="IEDName">
    <xs:simpleType>
      <xs:restriction base="xs:Name">
        <xs:enumeration value="IEDName"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:complexType>
<xs:complexType name="tHitem" mixed="true">
  <xs:complexContent>
    <xs:extension base="tAnyContentFromOtherNamespace">
      <xs:attribute name="version" type="xs:normalizedString" use="required"/>
      <xs:attribute name="revision" type="xs:normalizedString" use="required"/>
      <xs:attribute name="when" type="xs:normalizedString" use="required"/>
      <xs:attribute name="who" type="xs:normalizedString"/>
      <xs:attribute name="what" type="xs:normalizedString"/>
      <xs:attribute name="why" type="xs:normalizedString"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tVal">
  <xs:simpleContent>
    <xs:extension base="xs:normalizedString">
      <xs:attribute name="sGroup" type="xs:unsignedInt" use="optional"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```



```

    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="tValueWithUnit">
    <xs:simpleContent>
      <xs:extension base="xs:decimal">
        <xs:attribute name="unit" type="xs:token" use="required"/>
        <xs:attribute name="multiplier" type="tUnitMultiplierEnum" use="optional" default=""/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="tVoltage">
    <xs:simpleContent>
      <xs:restriction base="tValueWithUnit">
        <xs:attribute name="unit" type="xs:token" use="required" fixed="V"/>
        <xs:attribute name="multiplier" type="tUnitMultiplierEnum" use="optional" default=""/>
      </xs:restriction>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="tDurationInSec">
    <xs:simpleContent>
      <xs:restriction base="tValueWithUnit">
        <xs:attribute name="unit" type="xs:token" use="required" fixed="s"/>
        <xs:attribute name="multiplier" type="tUnitMultiplierEnum" use="optional" default=""/>
      </xs:restriction>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="tDurationInMilliSec">
    <xs:simpleContent>
      <xs:extension base="xs:decimal">
        <xs:attribute name="unit" type="xs:token" use="optional" fixed="s"/>
        <xs:attribute name="multiplier" type="tUnitMultiplierEnum" use="optional" fixed="m"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="tBitRateInMbPerSec">
    <xs:simpleContent>
      <xs:extension base="xs:decimal">
        <xs:attribute name="unit" type="xs:normalizedString" use="optional" fixed="b/s"/>
        <xs:attribute name="multiplier" type="tUnitMultiplierEnum" use="optional" fixed="M"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:schema>

```

**<CODE ENDS>**

### E.3 Substation syntax

Replace "2007B" with "2007B4".

### E.4 Data type templates

Replace "2007B" with "2007B4".

## E.5 IED capabilities and structure

Replace the existing text of Subclause E.5 with the following new text:

### SCL\_IED.xsd

This file contains most of the differences compared to the 2007B4 version. For backward compatibility it allows deprecated options, and sets defaults to instances from the 2003 A version.

#### <CODE BEGINS>

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:scl="http://www.iec.ch/61850/2003/SCL" xmlns="http://www.iec.ch/61850/2003/SCL"
xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="http://www.iec.ch/61850/2003/SCL"
elementFormDefault="qualified" attributeFormDefault="unqualified" version="Mixed.2007B4">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Informative general SCL schema for mixed systems version "2007" revision "B" release 4, for IEC 61850-6 Ed. 2.1.

      COPYRIGHT (c) IEC, 2016. All rights reserved. Disclaimer: The IEC disclaims liability for any personal injury,
      property or other damages of any nature whatsoever, whether special, indirect, consequential or compensatory, directly or
      indirectly resulting from this software and the document upon which its methods are based, use of, or reliance upon.
    </xs:documentation>
  </xs:annotation>
  <xs:include schemaLocation="SCL_BaseTypes.xsd"/>
  <xs:attributeGroup name="agAuthentication">
    <xs:attribute name="none" type="xs:boolean" use="optional" default="true"/>
    <xs:attribute name="password" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="weak" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="strong" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="certificate" type="xs:boolean" use="optional" default="false"/>
  </xs:attributeGroup>
  <xs:attributeGroup name="agSmvOpts">
    <xs:attribute name="refreshTime" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="sampleSynchronized" type="xs:boolean" use="optional" fixed="true"/>
    <xs:attribute name="sampleRate" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="dataSet" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="security" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="dataRef" type="xs:boolean" use="optional"/>
    <xs:attribute name="timestamp" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="synchSourceId" type="xs:boolean" use="optional" default="false"/>
  </xs:attributeGroup>
  <xs:attributeGroup name="agOptFields">
    <xs:attribute name="seqNum" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="timeStamp" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="dataSet" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="reasonCode" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="dataRef" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="entryID" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="configRef" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="bufOvfl" type="xs:boolean" use="optional" default="true"/>
    <xs:attribute name="segmentation" type="xs:boolean" use="optional" default="false"/>
  </xs:attributeGroup>
  <xs:attributeGroup name="agLDRef">
    <xs:attributeGroup ref="scl:agDesc"/>
    <xs:attribute name="iedName" type="tIEDName" use="required"/>
    <xs:attribute name="ldInst" type="tLDInst" use="required"/>
  </xs:attributeGroup>
  <xs:attributeGroup name="agLNRef">
    <xs:attributeGroup ref="agLDRef"/>
    <xs:attribute name="prefix" type="tPrefix" use="optional" default=""/>
    <xs:attribute name="lnClass" type="tLNClassEnum" use="required"/>
    <xs:attribute name="lnInst" type="tLNInstOrEmpty" use="required"/>
  </xs:attributeGroup>
  <xs:complexType name="tIED">
    <xs:complexContent>
      <xs:extension base="tUnNaming">
        <xs:sequence>
          <xs:element name="Services" type="tServices" minOccurs="0"/>
          <xs:element name="AccessPoint" type="tAccessPoint" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

        <xs:unique name="uniqueLNInAccessPoint">
            <xs:selector xpath="/scl:LN"/>
            <xs:field xpath="@inst"/>
            <xs:field xpath="@lnClass"/>
            <xs:field xpath="@prefix"/>
        </xs:unique>
    </xs:element>
    <xs:element name="KDC" type="tKDC" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
<xs:attribute name="name" type="tIEDName" use="required"/>
<xs:attribute name="type" type="xs:normalizedString" use="optional"/>
<xs:attribute name="manufacturer" type="xs:normalizedString" use="optional"/>
<xs:attribute name="configVersion" type="xs:normalizedString" use="optional"/>
<xs:attribute name="originalSclVersion" type="tSclVersion" use="optional" default="2003"/>
<xs:attribute name="originalSclRevision" type="tSclRevision" use="optional" default="A"/>
<xs:attribute name="originalSclRelease" type="tSclRelease" use="optional" default="1"/>
<xs:attribute name="engRight" type="tRightEnum" use="optional" default="full"/>
<xs:attribute name="owner" type="xs:normalizedString" use="optional"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tServices">
    <xs:all>
        <xs:element name="DynAssociation" type="scl:tServiceWithOptionalMax" minOccurs="0"/>
        <xs:element name="SettingGroups" type="scl:tSettingGroups" minOccurs="0"/>
        <xs:element name="GetDirectory" type="scl:tServiceYesNo" minOccurs="0"/>
        <xs:element name="GetDataObjectDefinition" type="scl:tServiceYesNo" minOccurs="0"/>
        <xs:element name="DataObjectDirectory" type="scl:tServiceYesNo" minOccurs="0"/>
        <xs:element name="GetDataSetValue" type="scl:tServiceYesNo" minOccurs="0"/>
        <xs:element name="SetDataSetValue" type="scl:tServiceYesNo" minOccurs="0"/>
        <xs:element name="DataSetDirectory" type="scl:tServiceYesNo" minOccurs="0"/>
        <xs:element name="ConfDataSet" type="scl:tServiceForConfDataSet" minOccurs="0"/>
        <xs:element name="DynDataSet" type="scl:tServiceWithMaxAndMaxAttributes" minOccurs="0"/>
        <xs:element name="ReadWrite" type="scl:tServiceYesNo" minOccurs="0"/>
        <xs:element name="TimerActivatedControl" type="scl:tServiceYesNo" minOccurs="0"/>
        <xs:element name="ConfReportControl" type="scl:tServiceConfReportControl" minOccurs="0"/>
        <xs:element name="GetCBValues" type="scl:tServiceYesNo" minOccurs="0"/>
        <xs:element name="ConfLogControl" type="scl:tServiceWithMaxNonZero" minOccurs="0"/>
        <xs:element name="ReportSettings" type="scl:tReportSettings" minOccurs="0"/>
        <xs:element name="LogSettings" type="scl:tLogSettings" minOccurs="0"/>
        <xs:element name="GSESettings" type="scl:tGSESettings" minOccurs="0"/>
        <xs:element name="SMVSettings" type="scl:tSMVSettings" minOccurs="0"/>
        <xs:element name="GSEDir" type="scl:tServiceYesNo" minOccurs="0"/>
        <xs:element name="GOOSE" type="scl:tGOOSECapabilities" minOccurs="0"/>
        <xs:element name="GSSE" type="scl:tServiceWithMax" minOccurs="0"/>
        <xs:element name="SMVsc" type="scl:tSMVsc" minOccurs="0"/>
        <xs:element name="FileHandling" type="scl:tFileHandling" minOccurs="0"/>
        <xs:element name="ConfLNs" type="scl:tConfLNs" minOccurs="0"/>
        <xs:element name="ClientServices" type="scl:tClientServices" minOccurs="0"/>
        <xs:element name="ConfLdName" type="scl:tServiceYesNo" minOccurs="0"/>
        <xs:element name="SupSubscription" type="scl:tSupSubscription" minOccurs="0"/>
        <xs:element name="ConfSigRef" type="scl:tServiceWithMaxNonZero" minOccurs="0"/>
        <xs:element name="ValueHandling" type="scl:tValueHandling" minOccurs="0"/>
        <xs:element name="RedProt" type="scl:tRedProt" minOccurs="0"/>
        <xs:element name="TimeSyncProt" type="scl:tTimeSyncProt" minOccurs="0"/>
        <xs:element name="CommProt" type="scl:tCommProt" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="nameLength" use="optional" default="32">
        <xs:simpleType>
            <xs:restriction base="xs:token">
                <xs:pattern value="32"/>
                <xs:pattern value="64"/>
                <xs:pattern value="6[5-9]"/>
                <xs:pattern value="[7-9]\d"/>
                <xs:pattern value="[1-9]\d\d+"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
</xs:complexType>
<xs:complexType name="tAccessPoint">
    <xs:complexContent>
        <xs:extension base="tUnNaming">
            <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="tServerAt"/>
</xs:choice>
<xs:element name="Services" type="scl: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="name" type="tAccessPointName" use="required"/>
<xs:attribute name="router" type="xs:boolean" use="optional" default="false"/>
<xs:attribute name="clock" type="xs:boolean" use="optional" default="false"/>
<xs:attribute name="kdc" type="xs:boolean" use="optional" default="false"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tCertificate">
  <xs:complexContent>
    <xs:extension base="tNaming">
      <xs:sequence>
        <xs:element name="Subject" type="tCert"/>
        <xs:element name="IssuerName" type="tCert"/>
      </xs:sequence>
      <xs:attribute name="xferNumber" type="xs:unsignedInt" use="optional"/>
      <xs:attribute name="serialNumber" use="required">
        <xs:simpleType>
          <xs:restriction base="xs:normalizedString">
            <xs:minLength value="1"/>
            <xs:pattern value="[0-9]+"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tCert">
  <xs:attribute name="commonName" use="required">
    <xs:simpleType>
      <xs:restriction base="xs:normalizedString">
        <xs:minLength value="4"/>
        <xs:pattern value="none"/>
        <xs:pattern value="CN=.+"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="idHierarchy" use="required">
    <xs:simpleType>
      <xs:restriction base="xs:normalizedString">
        <xs:minLength value="1"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:complexType>
<xs:complexType name="tServerAt">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:attribute name="apName" type="tAccessPointName" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tServer">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:sequence>
        <xs:element name="Authentication">
          <xs:complexType>
            <xs:attributeGroup ref="agAuthentication"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```



```

</xs:element>
<xs:element name="LDevice" type="tLDevice" maxOccurs="unbounded">
  <xs:unique name="uniqueLNInLDevice">
    <xs:selector xpath="/scl:LN"/>
    <xs:field xpath="@inst"/>
    <xs:field xpath="@lnClass"/>
    <xs:field xpath="@prefix"/>
  </xs:unique>
</xs:element>
<xs:element name="Association" type="tAssociation" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
<xs:attribute name="timeout" type="xs:unsignedInt" use="optional" default="30"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tLDevice">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:sequence>
        <xs:element ref="LN0"/>
        <xs:element ref="LN" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="AccessControl" type="tAccessControl" minOccurs="0"/>
      </xs:sequence>
      <xs:attribute name="inst" type="tLDInst" use="required"/>
      <xs:attribute name="ldName" type="tLDName" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tAccessControl" mixed="true">
  <xs:complexContent>
    <xs:extension base="tAnyContentFromOtherNamespace"/>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tAssociation">
  <xs:attributeGroup ref="agLNRef"/>
  <xs:attribute name="kind" type="tAssociationKindEnum" use="required"/>
  <xs:attribute name="associationID" type="tAssociationID" use="optional"/>
</xs:complexType>
<xs:element name="LN0">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="tLN0"/>
    </xs:complexContent>
  </xs:complexType>
  <xs:unique name="uniqueReportControlInLN0">
    <xs:selector xpath="/scl:ReportControl"/>
    <xs:field xpath="@name"/>
  </xs:unique>
  <xs:unique name="uniqueLogControlInLN0">
    <xs:selector xpath="/scl:LogControl"/>
    <xs:field xpath="@name"/>
  </xs:unique>
  <xs:unique name="uniqueGSEControlInLN0">
    <xs:selector xpath="/scl:GSEControl"/>
    <xs:field xpath="@name"/>
  </xs:unique>
  <xs:unique name="uniqueSampledValueControlInLN0">
    <xs:selector xpath="/scl:SampledValueControl"/>
    <xs:field xpath="@name"/>
  </xs:unique>
  <xs:key name="DataSetKeyLN0">
    <xs:selector xpath="/scl:DataSet"/>
    <xs:field xpath="@name"/>
  </xs:key>
  <xs:keyref name="ref2DataSetReportLN0" refer="DataSetKeyLN0">
    <xs:selector xpath="/scl:ReportControl"/>
    <xs:field xpath="@datSet"/>
  </xs:keyref>
  <xs:keyref name="ref2DataSetLogLN0" refer="DataSetKeyLN0">
    <xs:selector xpath="/scl:LogControl"/>
    <xs:field xpath="@datSet"/>
  </xs:keyref>
  <xs:keyref name="ref2DataSetGSELN0" refer="DataSetKeyLN0">

```

```
<xs:selector xpath="/scl:GSEControl"/>
<xs:field xpath="@datSet"/>
</xs:keyref>
<xs:keyref name="ref2DataSetSVLN0" refer="DataSetKeyLN0">
  <xs:selector xpath="/scl:SampledValueControl"/>
  <xs:field xpath="@datSet"/>
</xs:keyref>
<xs:unique name="uniqueDOIinLN0">
  <xs:selector xpath="/scl:DOI"/>
  <xs:field xpath="@name"/>
</xs:unique>
<xs:unique name="uniqueLogInLN0">
  <xs:selector xpath="/scl:Log"/>
  <xs:field xpath="@name"/>
</xs:unique>
</xs:element>
<xs:element name="LN" type="tLN">
  <xs:unique name="uniqueReportControlInLN">
    <xs:selector xpath="/scl:ReportControl"/>
    <xs:field xpath="@name"/>
  </xs:unique>
  <xs:unique name="uniqueLogControlInLN">
    <xs:selector xpath="/scl:LogControl"/>
    <xs:field xpath="@name"/>
  </xs:unique>
  <xs:key name="DataSetKeyInLN">
    <xs:selector xpath="/scl:DataSet"/>
    <xs:field xpath="@name"/>
  </xs:key>
  <xs:keyref name="ref2DataSetReport" refer="DataSetKeyInLN">
    <xs:selector xpath="/scl:ReportControl"/>
    <xs:field xpath="@datSet"/>
  </xs:keyref>
  <xs:keyref name="ref2DataSetLog" refer="DataSetKeyInLN">
    <xs:selector xpath="/scl:LogControl"/>
    <xs:field xpath="@datSet"/>
  </xs:keyref>
  <xs:unique name="uniqueDOIinLN">
    <xs:selector xpath="/scl:DOI"/>
    <xs:field xpath="@name"/>
  </xs:unique>
  <xs:unique name="uniqueLogInLN">
    <xs:selector xpath="/scl:Log"/>
    <xs:field xpath="@name"/>
  </xs:unique>
</xs:element>
<xs:complexType name="tAnyLN" abstract="true">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:sequence>
        <xs:element name="DataSet" type="tDataSet" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="ReportControl" type="tReportControl" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="LogControl" type="tLogControl" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="DOI" type="tDOI" minOccurs="0" maxOccurs="unbounded">
          <xs:unique name="uniqueSDI_DAIinDOI">
            <xs:selector xpath="/scl:DAI|/scl:SDI"/>
            <xs:field xpath="@name"/>
            <xs:field xpath="@ix"/>
          </xs:unique>
        </xs:element>
        <xs:element name="Inputs" type="tInputs" minOccurs="0">
          <!--<xs:unique name="uniqueExtRefInInputs">
            <xs:selector xpath="/scl:ExtRef"/>
            <xs:field xpath="@iedName"/>
            <xs:field xpath="@ldInst"/>
            <xs:field xpath="@prefix"/>
            <xs:field xpath="@lnClass"/>
            <xs:field xpath="@lnInst"/>
            <xs:field xpath="@doName"/>
            <xs:field xpath="@daName"/>
            <xs:field xpath="@intAddr"/>
          </xs:unique-->
        </xs:element>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

IECNORM © IEC. Click to view the full PDF of IEC 61850-6:2009/AMD1:2018

```

        <xs:element name="Log" type="scl:tLog" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="lnType" type="tName" use="required"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tLN">
    <xs:complexContent>
        <xs:extension base="tAnyLN">
            <xs:attribute name="prefix" type="tPrefix" use="optional" default=""/>
            <xs:attribute name="lnClass" type="tLNClassEnum" use="required"/>
            <xs:attribute name="inst" type="tLNInst" use="required"/>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="tLN0">
    <xs:complexContent>
        <xs:extension base="tAnyLN">
            <xs:sequence>
                <xs:element name="GSEControl" type="tGSEControl" minOccurs="0" maxOccurs="unbounded"/>
                <xs:element name="SampledValueControl" type="tSampledValueControl" minOccurs="0"
maxOccurs="unbounded"/>
                <xs:element name="SettingControl" type="tSettingControl" minOccurs="0"/>
            </xs:sequence>
            <xs:attribute name="lnClass" type="tLNClassEnum" use="required" fixed="LN0"/>
            <xs:attribute name="inst" type="xs:normalizedString" use="required" fixed=""/>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="tDataSet">
    <xs:complexContent>
        <xs:extension base="tUnNaming">
            <xs:choice maxOccurs="unbounded">
                <xs:element name="FCDA" type="tFCDA"/>
            </xs:choice>
            <xs:attribute name="name" type="tDataSetName" use="required"/>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="tFCDA">
    <xs:attribute name="ldInst" type="tLDInst" use="optional"/>
    <xs:attribute name="prefix" type="tPrefix" use="optional" default=""/>
    <xs:attribute name="lnClass" type="tLNClassEnum" use="optional"/>
    <xs:attribute name="lnInst" type="tLNInst" use="optional"/>
    <xs:attribute name="doName" type="tFullDOName" use="optional"/>
    <xs:attribute name="daName" type="tFullAttributeName" use="optional"/>
    <xs:attribute name="fc" type="tFCEnum" use="required"/>
    <xs:attribute name="fix" type="xs:unsignedInt" use="optional"/>
</xs:complexType>
<xs:complexType name="tControl" abstract="true">
    <xs:complexContent>
        <xs:extension base="tUnNaming">
            <xs:attribute name="name" type="tCBName" use="required"/>
            <xs:attribute name="datSet" type="tDataSetName" use="optional"/>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="tControlWithTriggerOpt" abstract="true">
    <xs:complexContent>
        <xs:extension base="tControl">
            <xs:sequence>
                <xs:element name="TrgOps" type="tTrgOps" minOccurs="0"/>
            </xs:sequence>
            <xs:attribute name="intgPd" type="xs:unsignedInt" use="optional" default="0"/>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="tTrgOps">
    <xs:attribute name="dchg" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="qchg" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="dupd" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="period" type="xs:boolean" use="optional" default="false"/>
    <xs:attribute name="gi" type="xs:boolean" use="optional" default="true"/>

```

```

</xs:complexType>
<xs:complexType name="tReportControl">
  <xs:complexContent>
    <xs:extension base="tControlWithTriggerOpt">
      <xs:sequence>
        <xs:element name="OptFields">
          <xs:complexType>
            <xs:attributeGroup ref="agOptFields"/>
          </xs:complexType>
        </xs:element>
        <xs:element name="RptEnabled" type="tRptEnabled" minOccurs="0"/>
      </xs:sequence>
      <xs:attribute name="rptID" type="tMessageID" use="optional"/>
      <xs:attribute name="confRev" type="xs:unsignedInt" use="required"/>
      <xs:attribute name="buffered" type="xs:boolean" use="optional" default="false"/>
      <xs:attribute name="bufTime" type="xs:unsignedInt" use="optional" default="0"/>
      <xs:attribute name="indexed" type="xs:boolean" use="optional" default="true"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tRptEnabled">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:sequence>
        <xs:element name="ClientLN" type="tClientLN" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="max" use="optional" default="1">
        <xs:simpleType>
          <xs:restriction base="xs:unsignedInt">
            <xs:minExclusive value="0"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tClientLN">
  <xs:attributeGroup ref="agLNRef"/>
  <xs:attribute name="apRef" type="tAccessPointName" use="optional"/>
</xs:complexType>
<xs:complexType name="tLogControl">
  <xs:complexContent>
    <xs:extension base="tControlWithTriggerOpt">
      <xs:attribute name="ldInst" type="tLDInst" use="optional"/>
      <xs:attribute name="prefix" type="tPrefix" use="optional" default=""/>
      <xs:attribute name="lnClass" type="tLNClassEnum" use="optional" default="LLN0"/>
      <xs:attribute name="lnInst" type="tLNInst" use="optional"/>
      <xs:attribute name="logName" type="tLogName" use="required"/>
      <xs:attribute name="logEna" type="xs:boolean" use="optional" default="true"/>
      <xs:attribute name="reasonCode" type="xs:boolean" use="optional" default="true"/>
      <xs:attribute name="bufTime" type="xs:unsignedInt" use="optional" default="0"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tInputs">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:sequence>
        <xs:element name="ExtRef" type="tExtRef" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tExtRef">
  <xs:attributeGroup ref="scl:agDesc"/>
  <xs:attribute name="iedName" type="tIEDNameOrRelative" use="optional"/>
  <xs:attribute name="ldInst" type="tLDInst" use="optional"/>
  <xs:attribute name="prefix" type="tPrefix" use="optional"/>
  <xs:attribute name="lnClass" type="tLNClassEnum" use="optional"/>
  <xs:attribute name="lnInst" type="tLNInst" use="optional"/>
  <xs:attribute name="doName" type="tFullDOName" use="optional"/>
  <xs:attribute name="daName" type="tFullAttributeName" use="optional"/>
  <xs:attribute name="intAddr" type="xs:normalizedString" use="optional"/>

```

```

<xs:attribute name="serviceType" type="tServiceType" use="optional"/>
<xs:attribute name="srcLDInst" type="tLDInst" use="optional"/>
<xs:attribute name="srcPrefix" type="tPrefix" use="optional"/>
<xs:attribute name="srcLNClass" type="tLNClassEnum" use="optional"/>
<xs:attribute name="srcLNInst" type="tLNInst" use="optional"/>
<xs:attribute name="srcCBName" type="tCBName" use="optional"/>
<xs:attribute name="pServT" type="tServiceType" use="optional"/>
<xs:attribute name="pLN" type="tLNClassEnum" use="optional"/>
<xs:attribute name="pDO" type="tFullIDName" use="optional"/>
<xs:attribute name="pDA" type="tFullAttributeName" use="optional"/>
</xs:complexType>
<xs:complexType name="tLog">
  <xs:complexContent>
    <xs:extension base="tUnNaming">
      <xs:attribute name="name" type="tLogName" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tControlWithIEDName">
  <xs:complexContent>
    <xs:extension base="tControl">
      <xs:sequence>
        <xs:element name="IEDName" minOccurs="0" maxOccurs="unbounded">
          <xs:complexType>
            <xs:simpleContent>
              <xs:extension base="tIEDName">
                <xs:attribute name="apRef" type="tAccessPointName" use="optional"/>
                <xs:attribute name="ldInst" type="tLDInst" use="optional"/>
                <xs:attribute name="prefix" type="tPrefix" use="optional"/>
                <xs:attribute name="lnClass" type="tLNClassEnum" use="optional"/>
                <xs:attribute name="lnInst" type="tLNInst" use="optional"/>
              </xs:extension>
            </xs:simpleContent>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="confRev" type="xs:unsignedInt" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tProtocol">
  <xs:simpleContent>
    <xs:extension base="xs:normalizedString">
      <xs:attribute name="mustUnderstand" type="xs:boolean" use="required" fixed="true"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="tGSEControl">
  <xs:complexContent>
    <xs:extension base="tControlWithIEDName">
      <xs:sequence>
        <xs:element name="Protocol" type="tProtocol" fixed="R-GOOSE" minOccurs="0"/>
      </xs:sequence>
      <xs:attribute name="type" type="tGSEControlTypeEnum" use="optional" default="GOOSE"/>
      <xs:attribute name="appID" type="tMessageID" use="required"/>
      <xs:attribute name="fixedOffs" type="xs:boolean" use="optional" default="false"/>
      <xs:attribute name="securityEnable" type="scl:tPredefinedTypeOfSecurityEnum" use="optional"
default="None"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="tSampledValueControl">
  <xs:complexContent>
    <xs:extension base="tControlWithIEDName">
      <xs:sequence>
        <xs:element name="SmvOpts">
          <xs:complexType>
            <xs:attributeGroup ref="agSmvOpts"/>
          </xs:complexType>
        </xs:element>
        <xs:element name="Protocol" type="tProtocol" fixed="R-SV" minOccurs="0"/>
      </xs:sequence>
      <xs:attribute name="smvID" type="tMessageID" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

```

<xs:attribute name="multicast" type="xs:boolean" default="true"/>
<xs:attribute name="smpRate" type="xs:unsignedInt" use="required"/>
<xs:attribute name="nofASDU" type="xs:unsignedInt" use="required"/>
<xs:attribute name="smpMod" type="tSmpMod" use="optional" default="SmpPerPeriod"/>
<xs:attribute name="securityEnable" type="tPredefinedTypeOfSecurityEnum" use="optional"
default="None"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tSettingControl">
<xs:complexContent>
<xs:extension base="tUnNaming">
<xs:attribute name="numOfSGs" use="required">
<xs:simpleType>
<xs:restriction base="xs:unsignedInt">
<xs:minInclusive value="1"/>
</xs:restriction>
</xs:simpleType>
</xs:attribute>
<xs:attribute name="actSG" use="optional" default="1">
<xs:simpleType>
<xs:restriction base="xs:unsignedInt">
<xs:minInclusive value="1"/>
</xs:restriction>
</xs:simpleType>
</xs:attribute>
<xs:attribute name="resvTms" type="xs:unsignedShort" use="optional"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tDOI">
<xs:complexContent>
<xs:extension base="tUnNaming">
<xs:choice minOccurs="0" maxOccurs="unbounded">
<xs:element name="SDI" type="tSDI">
<xs:unique name="uniqueSDI_DAIinSDI">
<xs:selector xpath="/scl:DAI/scl:SDI"/>
<xs:field xpath="@name"/>
<xs:field xpath="@ix"/>
</xs:unique>
</xs:element>
<xs:element name="DAI" type="tDAI"/>
</xs:choice>
<xs:attribute name="name" type="tDataName" use="required"/>
<xs:attribute name="ix" type="xs:unsignedInt" use="optional"/>
<xs:attribute name="accessControl" type="xs:normalizedString" use="optional"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tSDI">
<xs:complexContent>
<xs:extension base="tUnNaming">
<xs:choice minOccurs="0" maxOccurs="unbounded">
<xs:element name="SDI" type="tSDI"/>
<xs:element name="DAI" type="tDAI"/>
</xs:choice>
<xs:attribute name="name" type="tAttributeNameEnum" use="required"/>
<xs:attribute name="ix" type="xs:unsignedInt" use="optional"/>
<xs:attribute name="sAddr" type="xs:normalizedString" use="optional"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="tDAI">
<xs:complexContent>
<xs:extension base="tUnNaming">
<xs:sequence>
<xs:element name="Val" type="tVal" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
<xs:attribute name="name" type="tAttributeNameEnum" use="required"/>
<xs:attribute name="sAddr" type="xs:normalizedString" use="optional"/>
<xs:attribute name="valKind" type="tValKindEnum" use="optional"/>
<xs:attribute name="ix" type="xs:unsignedInt" use="optional"/>
<xs:attribute name="valImport" type="xs:boolean" use="optional"/>

```