it4power

ENTSO-E IST
Evaluation of export capabilities

Author	Document Version	Document Number
Christoph Brunner	V0.8	1401A03-011

Released	Last Date of Storage
[Name, Date]	20-7-2017

1 SCOPE

To support the engineering process, the IST shall be able to export the signal list in a format that can be easily consumed by an IEC 61850 system specification tool. This document discusses variants of formats for that export.

2 CONTEXT

The ENTSO-E IST (Interoperability Specification Tool) has two major purposes:
- By collecting the signal list for the various functions from all ENTSO-E members and mapping them on IEC 61850 data attributes, it's primary purpose is to identify possible gaps in the standard
- Providing a structured signal list with reference to IEC 61850 data models, it is an element in the overall IEC 61850 engineering process for the ENTSO-E members

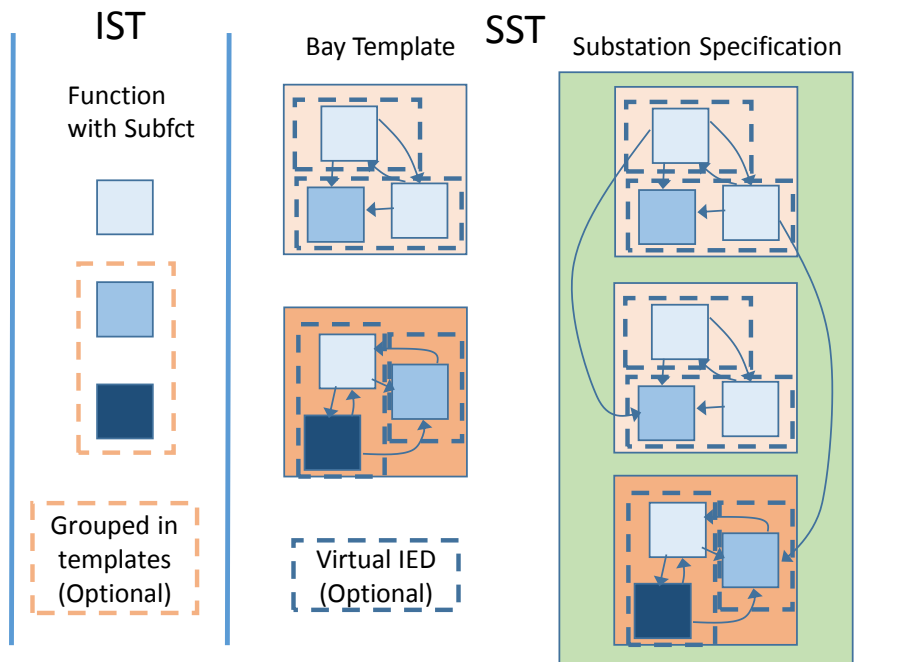A possible engineering process supported by the IST is shown in Figure 1

Figure 1 – Possible IEC 61850 engineering process using the IST

The IST provides the various functions with subfunctions and the associated output signals. In addition, it provides as well information about input signals to the functions. The functions may be grouped according to templates defined by the user.

It shall be possible to reuse the signal list for a typical function or multiple functions grouped in a template from the IST in the IEC 61850 engineering process. That process – as a top down process – starts with the preparation of a specification using an IEC 61850 system specification tool (SST).

The system specification in IEC 61850 is a concept that allows various levels of details. ENTSO-E is planning to support already as part of the specification a virtual design including signal flow specification to implement the protection and control schemes. Further, a two-step approach shall be supported by creating bay templates in a first step and then the substation specification out of the bay templates in a second step.

To support that process, the IST shall be able to export the signal list for one function or for multiple functions grouped in a template in a format that can be easily consumed by an SST.

3 IST AND CIM

The primary purpose of the ENTSO-E IST was to support IEC 61850 standardization and engineering process. In addition, it was as well planned that in a second step, the interactions between ENTSO-E IST and IEC 61968/61970 CIM shall be investigated.

In a similar way as the ENTSO-E profile is supporting the identification of gaps in IEC 61850 standard from an ENTSO-E perspective, it may as well be used to identify gaps in the CIM model. For that, a CIM export of the ENTSO-E profile may be required as well. This will however be investigated in a future step.

From the perspective of the engineering process, we currently do not see a need to export directly information from the ENTSO-E IST to CIM. The workflow for information out of the substation needed in the CIM world would be directly from IEC 61850 SCL through the IEC 61850 / CIM harmonisation work.

4 INFORMATION TO BE EXPORTED

This chapter describes the information from the IST that shall be exported to support the engineering process described in chapter 2.

As a minimum, the following information needs to be exported:

- Functions with subfunctions and its signals and settings as logical nodes and data objects; the function structure shall be visible (i.e. LNs that belong to one function shall be somehow grouped)
- Input signals to the subfunctions
- IEC 61850 version

It may as well be possible, to group the functions per domain. Optionally, the export may include the following, based on the level of specification added in the IST by the TSO:

- Identification of signals to be communicated to remote SCADA
- Communication services used to receive the inputs

Additionally, the following TSO specific information shall be included as well:

- Version identification of the templates
- TSO / Template identification
- TSO specific signal and function names

It shall preferably be possible to export a complete TSO template, to export multiple TSO templates (selectable which ones to export) in one file for documentation or to do the export by function as an individual file.

Note: as part of the discussion some possible new functional requirements for the IST tool were discussed:
- *Possibility to edit connections only on a single function*

- *Possibility to merge multiple TSO templates into a single one*

5 THE PROPOSED EXPORT FORMAT

For a detailed discussion providing the arguments for the solution, see Annex A. The format will use the following elements (sections) of SCL

- **Substation** to define domains, functions and subfunctions, the mapping of subfunctions to LNs and inputs to subfunctions
- **DataTypeTemplates** to define the outputs of the subfunctions as data objects of logical nodes

The proposed format will be a partially completed SSD file. It deviates from a fully defined SSD file in the following:

- It does not provide the real number of instances; it only includes functions and subfunctions that, depending on the content, may be part of a bay (type).
- It does not provide the full substation topology. As such it does not have any terminals of conducting equipment and it does not have any connectivity nodes.
- The data type templates are only specified down to the DO level. That means, we define LNTypes in the data type template section and list the DOs that are needed, but we do not specify a DOType. This is valid SCL for the specification.

The proposed format extends a valid SSD file with the following:

- Inputs with ExtRefs have been added as a private element at LNode level. The syntax of ExtRef is aligned with standard Ed 2.1 CDV SCL specification.

5.1 Basic elements of the SCL file

The following generic setup of the file shall be used:

- Attribute name of the substation element shall be "TEMPLATE"
- Attribute name of VoltageLevel element shall be "unspecified"
- Attribute name of bay shall be the name of the template
- For header, see clause 5.6

5.2 Mapping domains, functions and subfunctions

The hierarchical structure of the function definition in the IST will be mapped as follows (see as well Figure 3):

- Domains map on SCL "Function" elements within the bay element. The short name of the domain will map to the "type" attribute of the SCL "Function" element and the name to the "desc" attribute.
- Functions map on SCL "SubFunction" elements within the Function element specifying the domain . The short name of the function will map to the type attribute of the SCL "SubFunction" element and the name to the "desc" attribute.
- Subfunctions map on one of the following:
  o SCL "SubFunction" element within the SubFunction element specifying the function for all subfunctions that are not related to a LN for a

conducting equipment as shown in Figure 2. The short name maps to the "type" attribute and the long name to the "desc" attribute.

o SCL "EqFunction" element within a "ConductingEquipment" that is a child of the SCL "SubFunction" element specifying the function for all subfunctions related to conducting equipment. The short name maps to the "type" attribute and the long name to the "desc" attribute. The type of the conducting equipment depends on the LN and is shown in Figure 2.

LN	Type of Conducting equipment
XCBR	CBR
XSWI	DIS
TVTR	VTR
TCTR	CTR
YLTC	LTC

Figure 2 – Mapping of subfunctions related to conducting equipment

NOTE: A short name for the domain and a short name for the function are new elements to be added to the database of the IST.

SCL Element						Name	Type	desc		explanation
	Substation					TEMPLATE				
		VoltageLevel				unspecified				
			Bay			<TemplateName>				
				Function			*CTRL*	Control		IST "Domain"
					SubFunction		*BRK*	Breaker		IST "Function"
						ConductingEquipment	*CBR*	Breaker		
						EqFunction	*BIF*	Breaker Interface		IST "Subfunction"
							LNode	XCBR_BIF		
					SubFunction		*BC*	Breaker Control		IST "Subfunction"
						LNode	CSWI_BC			
					SubFunction		*SYC*	Synchrocheck		
						LNode	RSYN_SYC			IST "Subfunction"

Figure 3 – Mapping example

5.3 Defining output signals

The output signals of a function will be defined as data objects of the logical nodes. This is done in the data type template section of the SCL file. In the substation section, an LNode element is associated to the subfunction or to the EqFunction as shown in Figure 3. This LNode is of a particular type. The type name will be created as <LNClass>_<subfunctionShortName> (e.g. XCBR_BIF). Within the data type template section, that type is defined providing a list of data objects that belong to the logical node.

The signal name from the ENTSO-E profile shall be exported in the attribute "desc" of the DO element of the logical node type.

For the purpose of this export, the detailed structure of the data object will not be defined – so the specification is limited to define a logical node type with the data objects, but the data objects will not refer to a DOType.

5.4 Defining input signals

In order to define the input signals, an extension to the standard will be made. The extension is using the already existing element "ExtRef". According to the standard, this defines inputs to logical nodes in the context of an IED section. For this export format, the same elements will be used as inputs on the LNodes in the substation section as shown in Figure 4. To be compatible with standard SCL files, this extension will be placed in a private element according to IEC 61850-6. The type attribute of the private element shall be "ENTSO-E_IST"

SCL Element					Name	Type	desc	explanation
Substation					TEMPLATE			
	VoltageLevel				unspecified			
		Bay			\<template name\>			
			Function			*PROT*	Protection	IST "Domain"
				SubFunction		*DIS*	Distance Protection	IST "Function"
				SubFunction		*TL*	Trip Logic	IST "Subfunction"
					LNode	PTRC_TL		
					Private	ENTSO-E_IST		
					Inputs			
					ExtRef desc="Operate Zone 1"			
				SubFunction		*Z1*	Zone 1	IST "Subfunction"
					LNode	PDIS_Z1		
					Private	ENTSO-E_IST		
					Inputs			
					ExtRef desc="Current Samples"			
					ExtRef desc="Voltage Samples"			

Figure 4 – Extension of substation section to allow specification of inputs

5.5 Additional optional information to be included

The user while doing the export, can select the following optional information to be included (assuming that it has been specified in the IST)

- Identification of signals to be communicated to remote SCADA
- Communication services used to receive the inputs

To identify the signals that shall be communicated to the remote SCADA, a function "SCADA gateway" with a LN ITCI listing all the signals that shall be sent to the remote SCADA as inputs.

To identify the communication service to be used to receive the inputs, the ExtRef element shall support the attribute "pServT".

5.6 Additional TSO specific information to be included

The following TSO specific information will be included in the export:

- Version identification of the templates
- TSO / Template identification

The version identification of the template will be exported as "version" attribute of the header. The TSO and template identification will be concatenated in the "id" attribute of the header.

Additionally, the user can select to include the following information in the export:

- TSO specific signal and function names

TSO specific function and subfunction names shall be added to the attribute "desc" of the Function and SubFunction element of SCL substation section instead of the ENTSO-E function and subfunction name. The TSO specific signal names shall be exported with the attribute "desc" of the DO element in the logical node template instead of the ENTSO-E signal name.

Annex A - Discussion of possible solutions

A.1 DISCUSSION OF POSSIBLE SOLUTIONS

IEC 61850 does not define any particular inputs to an SST. In order to enhance the probability, that SST's available in the market can read the export from the IST, the file format should be as close as possible to a format, an SST most likely will understand which is the ssd file produced by the SST. **Error! Reference source not found.** shows the various elements within an SCL file.
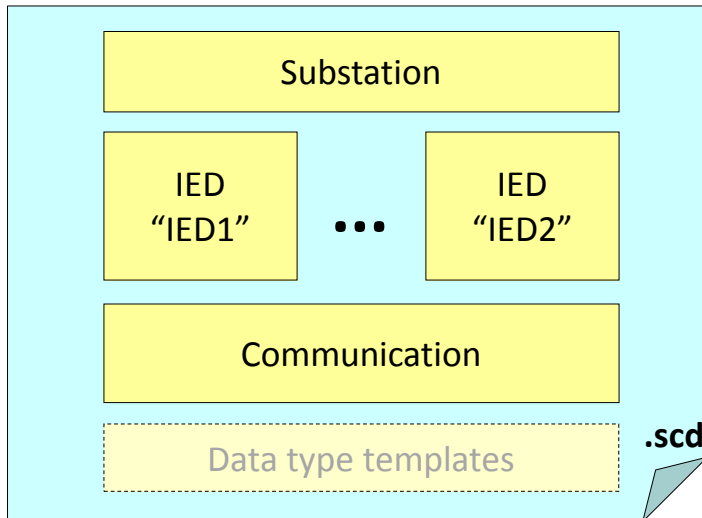

Figure 5 – The elements of the SCL file

In IEC 61850 SCL, a data model (and hence a signal list as exported from the IST) is mostly defined in the data type template section. It is in the data type template section that the details of a logical node – data objects and their data attributes – are specified. The data type template section is referred by either a LN in the IED section or an LNode in the substation section.

The export needs to be able to support the following two major aspects:

- Export the functional structure (functions / subfunctions) with the detailed signals
- Export the declaration of unbound inputs

NOTE: the inputs declared are unbound, since they are not yet in the context of an instantiated system and hence cannot refer to concrete signals.

While the IED section allows the definition of unbound input signals through ExtRef's, the substation section allows the structuring of the LNs in functions and subfunctions. Additionally, the IED section is typically not present in an SSD file and may not be supported by an SST.

So basically, we have three obvious possibilities for the export that are described in more details further in this document:

- Use the substation section and add unbound signals as private extensions or as InRef DOs with only a description (this is the preferred approach for rte)
- Use the IED section and map the function / subfunction structure on a hierarchy of logical devices
- Use both and create the links between them

A.2 USE OF THE SUBSTATION SECTION

The substation is basically the part of the SCL that is filled in by a system specification tool. It defines the single line diagram and the associated functional requirements.

Edition 2.1 of IEC 61850 added some extensions to the specification of the substation section to support a function oriented structure. Using that, an export of from the IST could be using the approach illustrated in Figure 6.

SCL Element							Name	Type	Explanation
Substation							AA1		
	VoltageLevel						E1		
		Bay					Q3		
			Function					*PROT*	Protection
				SubFunction				*DIS*	Distance Protection
					SubFunction			*TL*	Trip Logic
						LNode		PTRC	
					SubFunction			*ZONE*	Zone 1
						LNode		PDIS1	
					SubFunction			*ZONE*	Zone 2
						LNode		PDIS2	

Figure 6 – Using the substation section for the export

With that approach, we would have the following mapping of the IST elements:

- The IST domain will be mapped on an SCL function
- The IST function will be mapped on an SCL SubFunction
- The IST subfunction will be mapped on a LN in an SCL SubFunction that is hierarchically a level below the SubFunction where the IST function is mapped
- The various output signals and settings would be available through the details of the logical node specified in the data type template section.

NOTE: the additional level of SubFunction is required from an SCL schema perspective, since multiple LNs of the same class are not allowed in the same SubFunction.

To define the inputs with that approach, we basically have two options:

- Request a standard extension to support the declarations of inputs to logical nodes in the substation section in a similar way as it is possible today in the IED section using ExtRefs.
- Use the data objects "InRef" of the logical nodes. These data objects are foreseen to document in the run time data model input references; they can be preconfigured in the data type template sections with values. However, InRef use IED specific references, so this can as well not be done in the substation section only without standard extension.

Since at this stage we deal with unbound inputs, we can for both approaches not fill in a complete reference. With ExtRef we can use the same approach as done by an IED tool in an ICD/IID file – specify the LN class and DO to indicate the expected CDC; a partial reference is not possible with InRef. In addition, in both cases we can use a description that explains the semantics of the input.

A.3 USE OF THE IED SECTION

When using the IED section, a particular IED kind would need to be specified in the standard (e.g. SPECIFICATION instead of TEMPLATE what we have in an icd file). An export could then be made as shown in Figure 7.

SCL Element						Name	Type	Explanation
IED						SPEC		
	AccessPoint							
		Server						
			LogicalDevice					Protection
			Logical Device					Distance Protection
			LN				PTRC	Trip Logic
			LN				PDIS1	Zone 1
			LN				PDIS2	Zone 2
			LN				PDIS3	Zone 3

Figure 7 – Using the IED section

With that approach we would have the following mappings:

- The IST domain is mapped on a higher level logical device
- The IST function is mapped on a lower level logical device
- The IST subfunction is mapped on a logical node
- As with the version of the mapping to the substation section, the details of the signals and settings is available in the data type template section.

To define the inputs when using the IED section, we can use the approach that has recently be agreed to declare unbound inputs for a LN in an IED. This is illustrated in **Error! Reference source not found.**. It is possible to declare the semantic of the input in the description as well as the expected data type by referring to a DO
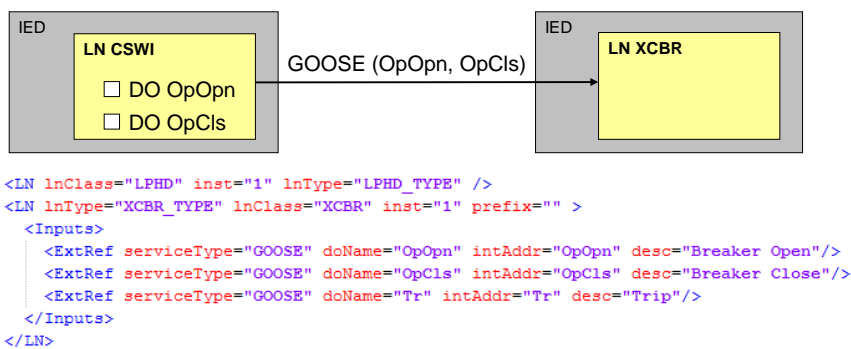


```
<LN lnClass="LPHD" inst="1" lnType="LPHD_TYPE" />
<LN lnType="XCBR_TYPE" lnClass="XCBR" inst="1" prefix="" >
  <Inputs>
    <ExtRef serviceType="GOOSE" doName="OpOpn" intAddr="OpOpn" desc="Breaker Open"/>
    <ExtRef serviceType="GOOSE" doName="OpCls" intAddr="OpCls" desc="Breaker Close"/>
    <ExtRef serviceType="GOOSE" doName="Tr" intAddr="Tr" desc="Trip"/>
  </Inputs>
</LN>
```

Figure 8 – Declaration of inputs in an IED section

A.4 COMBINED APPROACH

Apparently, the use of the substation section has the advantage to be in the scope of a specification and hence having a high probability to be understood by a SST. However today there is not standardized agreement, how to specify input signals for that. This is available in the IED section. So it could be considered to use a combined approach:

- Use the substation section for the basic export of the functions, subfunctions and signals
- Use the IED section for the additional export of input signals

While using that combined approach, the link between the elements in the substation section and the elements in the IED section would be done the same way as in a fully configured SCD file where the IEDs functions are bound to the primary process equipment. This is shown in **Error! Reference source not found.**.
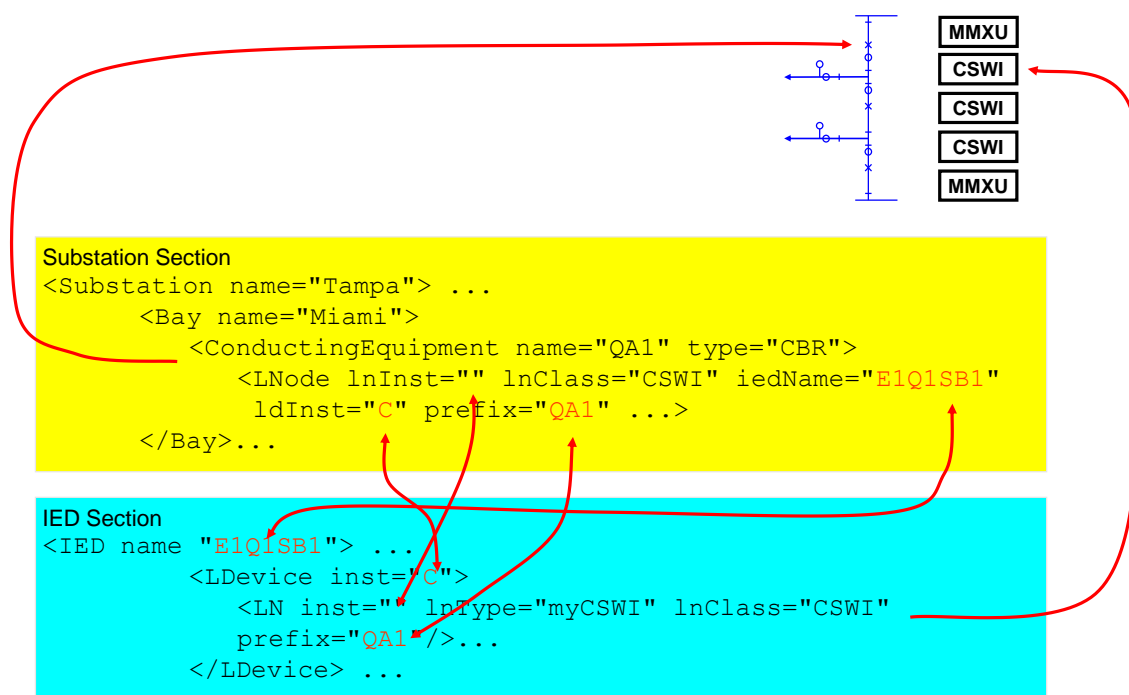

Figure 9 – Linking the IED section to the substation section

A.5 COMPARISON OF VARIANTS

The substation section is the place that describes the process in an implementation independent form. So this seems to be the natural place to export the IST content. However, there are a few issues that ENTSO-E needs to be aware:

- Declaration of inputs as part of the functional specification through a substation section is today not supported by the standard.
- There is ongoing work to standardize on function / subfunction types. In the future, the use of these standardized types would be expected. Today, ENTSO-E would use the function / subfunctions as they have been defined as part of the ENTSO-E profile in the IST.
- Additionally, the structure would allow for more hierarchical levels.
- So basically, once the standardization work on function and subfunctions is complete, there may be additional constraints that have to be considered, when mapping the content of the IST to IEC 61850 SCL substation section.

While the IED section already has today the possibility to declare inputs, it is unlikely that a SST will understand the IED section. Even if in the future an "IST" file will be available to declare the requirements on an IED, this should not be mandatory for a specification.

As a conclusion, it is recommended to use the substation section only and make extensions to the standard as required.

To support the specification of inputs, it is recommended to extend SCL using ExtRef as part of the LNode element and not InRef in the logical node declaration.