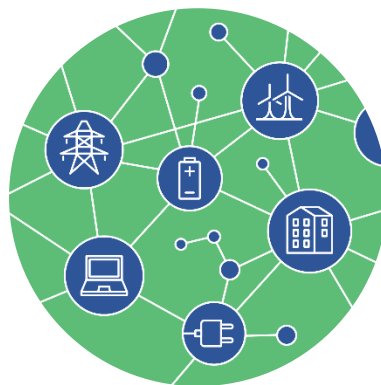




**OPTIMAL SYSTEM-MIX OF FLEXIBILITY
SOLUTIONS FOR EUROPEAN ELECTRICITY**

Recommendations for IEC61850 WG10 and the industry

WP7.1 - Deliverable 7.3



Contact: www.osmose-h2020.eu



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 773406

Document properties

Project Information

Programme	Optimal System-Mix Of Flexibility Solutions For European Electricity
Project acronym	OSMOSE
Grant agreement number	773406
Number of the Deliverable	7.3
WP/Task related	Task 7.1

Document information

Document Name	Recommendations for IEC61850 WG10 and the industry
Date of delivery	28/02/2022
Status and Version	Version 3.1
Number of pages	55

Responsible

Document Responsible	Ch. Brunner (it4power) / T. Sterckx (Elia)
Author(s)	Ch. Brunner (it4power) / T. Sterckx (Elia)
Reviewer(s)	Nathalie Grisey (RTE) / Gorazd Azman (ELES)
Approver	Nathalie Grisey (RTE)

Dissemination Level

Type (distribution level)	<input checked="" type="checkbox"/> PU, Public <input type="checkbox"/> CO – full consortium, Confidential, only for members of the consortium (including the Commission Services) <input type="checkbox"/> CO – some partners, Confidential, only for some partners (list of partners to be defined)
---------------------------	---

Review History

Version	Date	Reviewer	Comment
V3.0	28.02.2022	Nathalie Grisey	Final review finished - Publication

Table of content

0	Executive summary	1
1	List of acronyms and abbreviations	2
2	Introduction	4
2.1	Engineering process as defined by IEC61850	4
3	Engineering process developed in Osmose.....	5
3.1	SCL Extensions required to support the process.....	8
3.2	Standardisation	9
3.2.1	IST Export format	9
3.3	Specification.....	10
3.3.1	Importing IST files in System Specification Tools	11
3.3.2	Documenting a single line in the .ssd file.....	16
3.3.3	Instantiating functions and subfunctions.....	16
3.3.4	Creating dataflow between subfunctions	18
3.3.5	Allocation of subfunctions to virtual IEDs.....	22
3.3.6	Finalized ISD/SSD files	23
3.4	IED selection and Procurement.....	23
3.4.1	IED selection.....	24
3.4.2	Creating an extended ICD with enhanced substation section	25
3.4.3	Selection of IEDs based on a specification.....	26
3.5	Configuration.....	29
3.6	Commissioning.....	31
3.7	User right management during the engineering process	31
4	SCL file content	32
4.1	ISD/ICD content	32
5	Comparison between SCL files	36
6	Engineering process for dataflow between 2 substations	37
7	Documentation	40
8	Testing and Simulation.....	40
9	Modelling.....	41
9.1	Battery data modelling.....	41
9.1.1	Operational functions	44
9.1.2	Power management function – LN DPMC	45

9.1.3	The resource LNs.....	46
9.2	Demonstrator data modelling gaps.....	46
9.2.1	Autorecloser use case.....	46
9.3	Additional modelling recommendations	52
9.3.1	Modelling of physical resources	52
9.3.2	Teleprotection GOOSE tunnelling	53
10	Annexes	53
10.1	A: Interoperability test specification (ID7.2.A)	53
10.2	B: SCL comparison specification (ID7.3.B)	53
10.3	C: Enhancements of SCL for virtual signal flow and virtual IED (ID7.3.C).....	53
10.4	D: Annex D - IEC_61850-6-100.XSD.2019A6.WD.2021-08-24 (ID7.3.D)	53
11	Overview of all recommendations.....	54

List of figures

Figure 1 - the IEC61850 engineering process (From IEC61850-6:2009 + AMD1:2018).....	5
Figure 2 - Osmose engineering process.....	7
Figure 3 - Extended IEC61850 engineering process with elements from Osmose.....	8
Figure 4 – system specification	11
Figure 5 - Osmose function templates	12
Figure 6 – function template versioning	12
Figure 7 - GUID example.....	13
Figure 8 - Function template validation example.....	14
Figure 9 - Function template validation example.....	14
Figure 10 - Namespace selection example.....	15
Figure 11 – Enumerated element missing different values	15
Figure 12 – Enumerated values added by SST	16
Figure 13 - Osmose single line example.....	16
Figure 14 - Function example with inputs	17
Figure 15 - Power System Resource example.....	18
Figure 16 - PowerSystemResource SCL example.....	18
Figure 17 - Example of SCL specification and implementation of extRefAddr.....	20
Figure 18 – Example of extRefAddr allowing documentation of deviation in receiving LN and function LN.....	21
Figure 19 - ICD creation	26
Figure 20 - Selection of IEDs in the SCT	27
Figure 21 - creating the SCD file	30
Figure 22 - Creating IED configuration	31
Figure 23 - Comparison process steps	36
Figure 24 - Usecase for intersubstation dataflow	37
Figure 25 – SSD/SED exchange process without virtual IEDs.....	38
Figure 26 – SSD/SED exchange process with virtual IEDs.....	39
Figure 27 - Energy flow between substation A and B	42
Figure 28 - Energy flow regulation principle (from IEC61850-7-420).....	43
Figure 29 - Storage management system dataflow.....	44
Figure 30 - Use case description	47
Figure 31 - Variant 1.....	49
Figure 32 - Variant 2.....	50
Figure 33 - Variant 3.....	51
Figure 34 - Variant 4.....	52

List of tables

Table 1: Abbreviations and technical words.....	3
Table 2 - Actors in process steps	6

0 Executive summary

The Subtask 7.1.3 aims at providing a series of recommendations for the energy market, with a focus on digital substation engineering. Parties involved are:

- IEC61850 WG10 (standardization)
- Engineering tool suppliers
- IED suppliers
- Users (digital substation users (from the industry, utilities))

Recommendations apply to the domains of improvement addressed within the Osmose project, more specifically:

- Datamodel gaps
- Engineering process
 - o SCL Namespace
 - o SCL file content
 - o Engineering usecases and process steps
 - o SCL file comparison during selection and procurement stage

This document aims to address the points identified as next steps in deliverable 7.1, related to the engineering process and input for standardisation (WG10 IEC61850).

The framework in this document are the different steps of the engineering process, completed with additional chapters related to SCL file content, comparison, testing and simulation and data modelling.

The items described in this document are demonstrated and documented in deliverable 7.2 of Osmose which is focussing on the execution and evaluation of the process demonstrator.

Readers of this document should have knowledge of the IEC61850 standard and a basic understanding of its configuration language SCL, and as well have read deliverable 7.1.

1 List of acronyms and abbreviations

This table lists the acronyms and technical words used in this document.

Acronym	Meaning
CID	Configured IED Description
Data profile	Group of functions and subfunctions defined by an end-user representing all applications of this user
DB	Data Base
DO	Data Object
DA	Data Attribute
DER	Distributed Energy Resource
Domain	A domain represents a group of functions within a specific protection or automation domain (as introduced by ENTSO-E IST)
End-User	Party responsible for exploiting the substation after a project is finished
ENTSO-E	European Network of Transmission System Operators for Electricity
ENTSO-E profile	Profile defined by the ENTSO-E IEC61850 taskforce, containing a data profile and engineering process concept
FSD	Function specification description
Function	A function represents a group of subfunctions (as introduced by ENTSO-E IST)
GUID	Globally unique identifier
GOOSE	Generic Object Oriented Substation Event
ICD	IED configuration description
ICT	IED configuration tool
IED	Intelligent electronic device
IID	Instantiated IED description
ISD	IED Specification Description
IST	Interoperability specification tool (developed and hosted by ENTSO-E)
LN	Logical Node
NSD file	Namespace Description
SA system	Substation Automation system
SCD	System configuration description

SCL	Substation Configuration Language
SCT	System configuration tool
SED	System exchange description
SPACS	Substation Protection Automation and Control System
SSD	System specification description
SST	System specification tool
STT	System testing tool
subfunction	A subfunction (as introduced by ENTSO-E IST) represents a functional building block of a substation protection and automation system, and contains one or more output and input signals (Output signals are mapped to an IEC61850-7-4 datamodel).
template	A container of a subset of functions and subfunctions,
User	Any party executing the engineering process, this can be for example a TSO or a system integrator
Vendor	Vendor of substation protection and automation IEDs
WG10	TC57 workgroup related to the development and maintenance of the IEC61850 standard
IEC61850	IEC61850 is an international standard defining Communication Networks and Systems for Power Utility Automation
IEC61850-6-100	SCL Function Modelling for Substation Automation (in preparation)
IEC61850-7-4	Basic communication structure for substation and feeder equipment – Compatible logical node classes and data classes

Table 1: Abbreviations and technical words

2 Introduction

The Osmose project aims at improving the IEC61850 engineering process in order to cover steps from specification to commissioning in the most efficient way, with guarantee of a high quality, taken into account the needs of users and vendors. The process is demonstrated and documented in deliverable 7.2 and this document aims to provide the conclusions and recommendations to the industry (standard and market).

In deliverable 7.1, the following actions were identified to be addressed by the subtasks 7.1.2 and 7.1.3:

1. Define the necessary SCL extensions required to execute the engineering process
2. Define the ENTSO-E profile enhancements and extension regarding gaps and energy storage
3. Implement this process in the participating engineering tools
4. Demonstrate the full process through a demonstrator setup
5. Provide input for standardization in IEC61850 WG10

This document aims to provide an answer to topics 1,2 and 5.

The backbone for this document will be the engineering process, with detailed documentation of each step, referring to a series of annex documents providing details on specific topics.

2.1 Engineering process as defined by IEC61850

The IEC61850 part 6 Edition 2.1 defines an engineering process, focussing on system engineering and configuration.

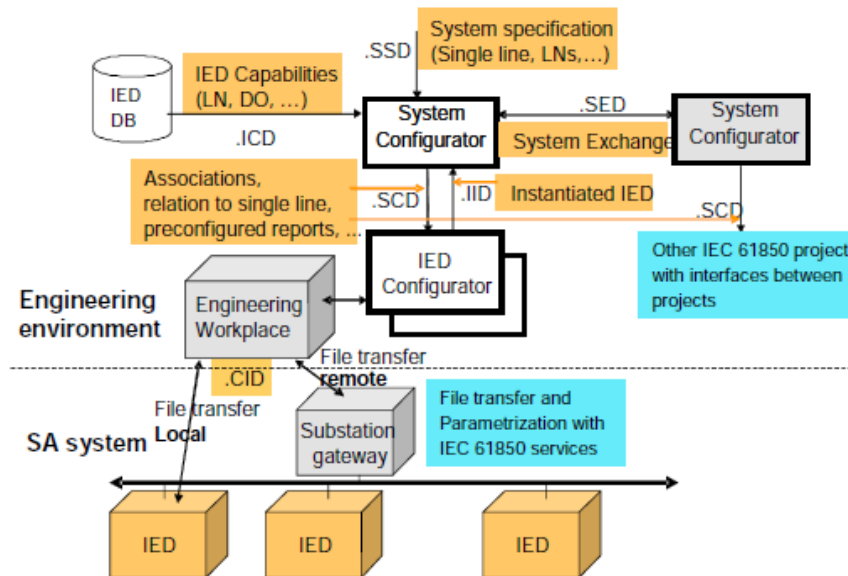


Figure 1 - the IEC61850 engineering process (From IEC61850-6:2009 + AMD1:2018)

This process is used as a starting point, and extensions are developed within Osmose. Focus of the extensions are on implementing

- creating vendor (and IED) independent specification
- specification of dataflow
- use of a user specific profile (function templates)
- traceability of the specification throughout the engineering process

The extensions, as well as other related improvements are all part of a series of recommendations to the IEC61850 standard, and will mostly be integrated in the new IEC61850-6-100.

3 Engineering process developed in Osmose

The engineering process developed and demonstrated in Osmose consists of 5 different stages:

- **Standardisation:** representing the concept and profile applied by the user (may rely on an external profile like ENTSO-E profile)
- **Specification:** creating a vendor independent project or template specification
- **Procurement:** selecting the devices from the vendor(s) corresponding to the specification needs
- **Configuration:** Integrating the selected devices within the specification and selected devices to build a system configuration
- **Commissioning:** creating IED configuration files and executing the commissioning

Within this process 2 variants are identified for the specification step:

- **Specification containing virtual IEDs:** the user wants to specify the physical allocation of functions to IEDs and wants to communicate IED requirements via a dedicated SCL file for a specific IED
- **Specification without virtual IEDs:** the user creates a functional specification without allocation of functions to IEDs and relies on the vendor execute the most optimal allocation related to the technical capabilities of his devices

Within the process different roles exists:

- **User:** end user of the system, responsible for the correct execution of the engineering process. This might be a user from a utility or the industry.
- **Vendor:** providing an offer for his devices and helping the user to create a system and IED configuration
- **Integrator:** partner contracted by the user to execute 1 or more tasks of the engineering process.
- **ENTSO-E:** providing the ENTSO-E profile datamodel (library of functions/subfunctions). It is considered a user may also create and use its own profile.

Each actor may (optionally¹) participate in selected number of process steps.

Process step	Actor			
	User	Vendor	Integrator	ENTSO-E
Standardisation	X			(X)
Specification	X			
Procurement	X	X		
Configuration	X		(X)	
Commissioning	X		(X)	

Table 2 - Actors in process steps

¹ Referred to by '(X)' in the table

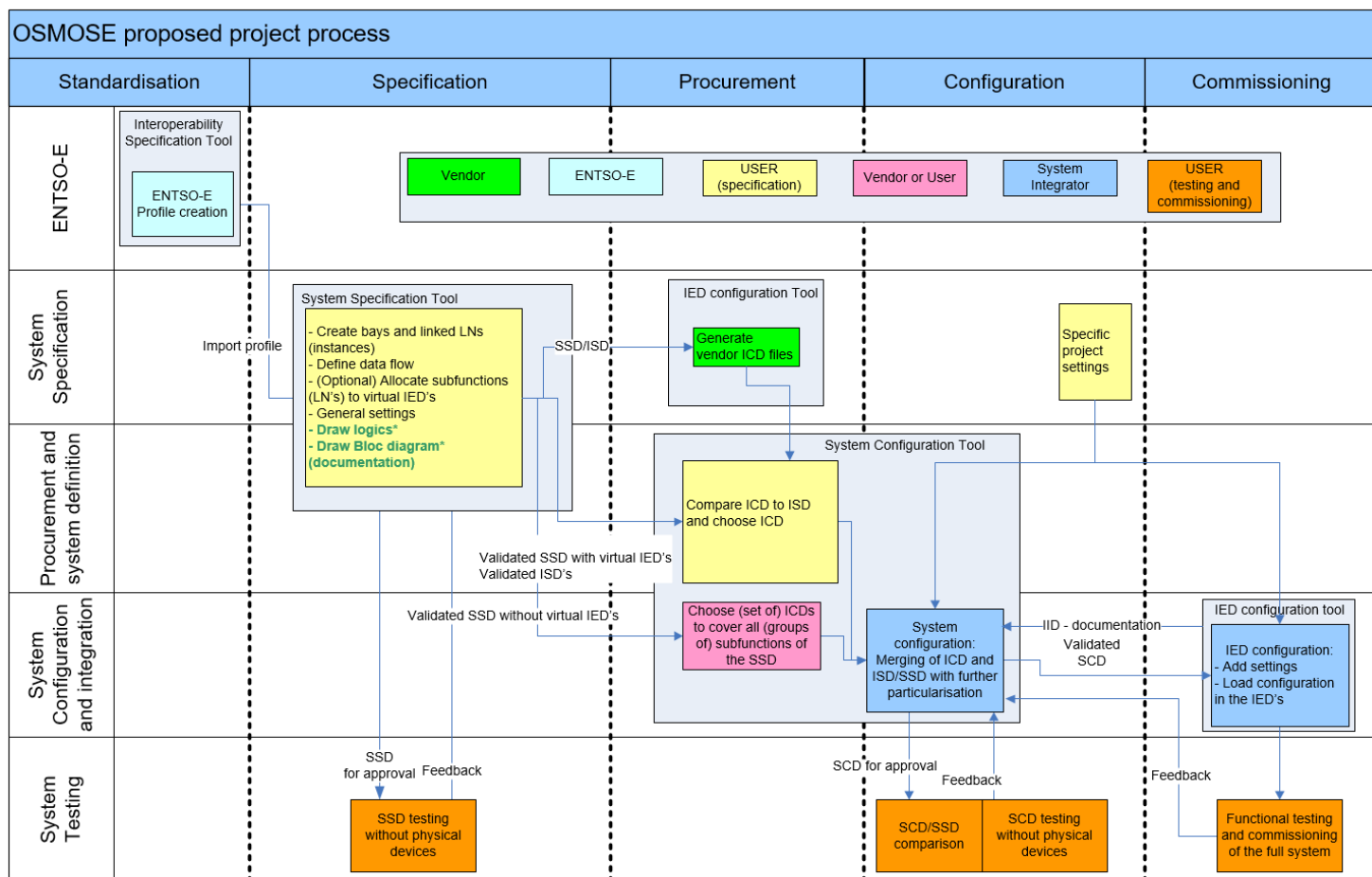


Figure 2 - Osmose engineering process

In order to show how this process compares to the original process from IEC61850-6, some extensions can be shown.



In the following chapters, each step of the process is explained and documented, including the recommendations for the IEC61850 standard working group² linked to each step.

In order to support the engineering process, extensions to the IEC61850 SCL language are required. In particular related to:

- ² Also referred to as WG10

The details of the SCL extensions defined within Osmose are defined in 10.3 Annex C: Enhancements of SCL for virtual signal flow and virtual IED (ID7.3.C). Additionally, the developed namespace is attached as 10.4 Annex D: Annex D - IEC_61850-6-100.XSD.2019A6.WD.2021-08-24 (ID7.3.D)

It is recommended for WG10 to include these enhancements in the IEC61850-6-100 technical report. It is recommended for IEC61850 engineering tools to support these SCL extensions in order to support the engineering process as described in this document.

3.2 Standardisation

The standardisation step consists of creating the basic block templates required for the definition of the specification. These templates are called .fsd template files and contain the definition of a function (optionally containing one or more subfunctions) mapped to LNode(s). Optionally a subfunction may contain predefined inputs to include placeholders for future dataflow definition. Templates may be created by the user, but may also be imported from an external user profile (for example from ENTSO-E or a specific utility (TSO/DSO)).

.fsd files may be created by dedicated tools (interoperability specification tool) or as an integrated functionality in system specification tools³.

3.2.1 IST Export format

It is important that the .fsd files are created in a way that instantiation in the specification step is possible, therefore the export format needs to be defined. Chapter 10.3 Annex C: Enhancements of SCL for virtual signal flow and virtual IED (ID7.3.C) provides the full documentation of this export format.

The export elements described in the annex are

- Functions with subfunctions and their signals and settings as Logical Nodes and Data Objects
- Input signals to the subfunctions
- IEC 61850 version of the data model
 - It is recommended for tools to provide in the DataTypeTemplate section the used datamodel namespace for each LNode. If not present in the .fsd the SST should update it during .fsd import

Within the Osmose project, the .fsd files are created with the extension .ssd for tool compatibility reasons.

³ It is recommend for system specification tools to integrate this functionality

It is recommended for WG10 to define the .fsd concept and format and for engineering tools to implement the required functions for creating and managing .fsd files.

It is recommended for function templates, to place the function definition under an SCL element called “Process”, to create a generic location and allow instantiation at different substation elements⁴.

It is recommended for WG10 to provide clear definitions for (sub)function attributes. As a recommendation for use inside the .fsd

- Name: to be used to identify the function template (in the .fsd). Will be modified when instantiated
- Type: identical to name (in .fsd). Will represent the reference of the template in instantiated functions (not modified in instances)
- Text: free text to be given by the user as detailed description of the function. For tool implementation reasons, it is not recommended to use it for transferring information to function instances
- Desc : should be used to give a hint to the user about the function definition

3.3 Specification

In the specification step the different function template description building blocks are instantiated in order to create a specification file .ssd (System Specification Description). This file may be created for a specific substation project or as part of a set of frame agreement template files. This step is executed in system specification tools⁵ (SST).

⁴ Not implemented in Osmose (all functions placed under a bay “TEMPLATE”)

⁵ It is recommended for system specification tools to support the specification process step as described in this document

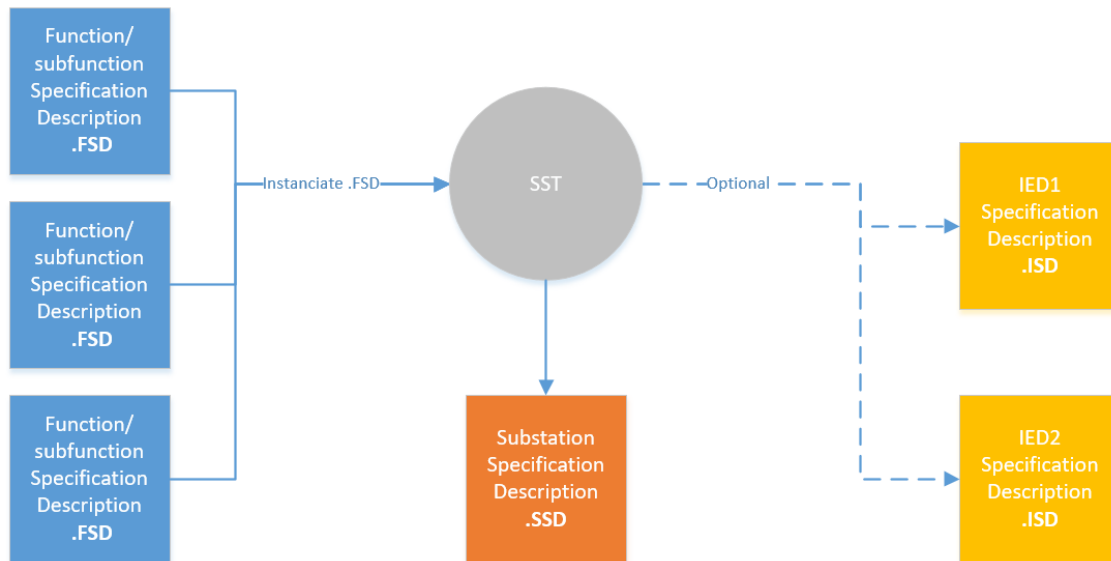


Figure 4 – system specification

Additionally, some optional elements may be added to complete the specification

1. Elements representing the primary equipment (conducting elements) allowing to document the single line diagram (with optionally included coordinate attributes)
2. Physically allocating the functions and subfunctions to IEDs, if the user wants to specify IED requirements containing the required functions and subfunctions. The IEDs part of the specification are called virtual IEDs and may be introduced
 - a. Only for physical allocation of subfunctions⁶
 - b. For a full IED specification represented by ISD files, including its datamodel and additional requirements exposed to other IED section related elements (for example “services”)⁷
3. Specification of selected element values related to the applied use case, exposed as new SCL elements DOS and DAS. These elements then overrule any DA/BDA elements from the LNode datatype template.
4. Documentation of the behaviour of a (sub)function or a group of (sub)functions⁸. This may be a graphical, logical (conform IEC61131-3) or textual description.

Detailed steps of the specification process related to tool interoperability testing are described in Annex A – Test Case 10: project specification

3.3.1 Importing IST files in System Specification Tools

⁶ Not implemented in Osmose: it is recommended for WG10 to provide ways in SCL to allow this

⁷ It is recommended for WG10 to introduce the virtual IED concept as demonstrated in Osmose

⁸ Not implemented in Osmose: it is recommended to WG10 to provide a way to allow linking this documentation to the SCL specification

It is recommended for WG10 to provide a way to keep track in the specification of the instantiated function templates, including their version. The introduction of a GUID element for SCL files is a solution that should be investigated⁹. An example of an implementation with GUID of the .fsd in an .ssd file is shown below.

```
<Function name="Pushbutton" desc="">
  <Private type="eIEC61850-6-100">
    <!-- GUID of the FTD file representing all push buttons -->
    <eTr_6-100:FtdReference guid="123e4567-e89b-12d3-a456-426655440001" version="1" revision="3"/>
  </Private>
</Function>
```

Figure 7 - GUID example

3.3.1.2 Data Type Correction and Completion

During import the Data Type Templates of the function templates need to be merged with the Data Types of the project (coming from other function templates). Collisions are handled through renaming by SST. In case of the Data Types are wrong or incomplete regarding SCL schema, they need to be corrected or completed by the SST¹⁰ (For example, choice of ENUM values). SSD and ISD that are created with imported function template instances should optionally set the NamPIt.InNs based on the used Namespace (NSD file). SST interprets the data types as the minimum set of data to be used in the specification dataflow.

Remark: The NSD validation is not taken into account for the specification, allowing a user to restrict the LNodeType to what is required in his own context (mandatory DO in NSD could be removed).

3.3.1.2.1 Example of incompleteness of the imported Data Types

Enum definitions of ENS_CBIF are missing

⁹ Not implemented in Osmose project

¹⁰ Recommendation for SST - Done manually in Osmose project

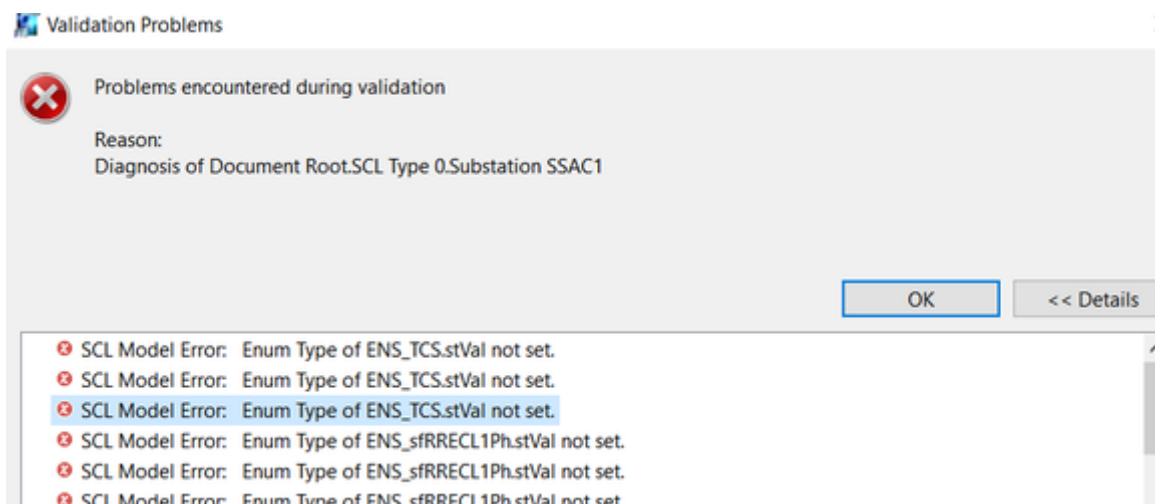


Figure 8 - Function template validation example

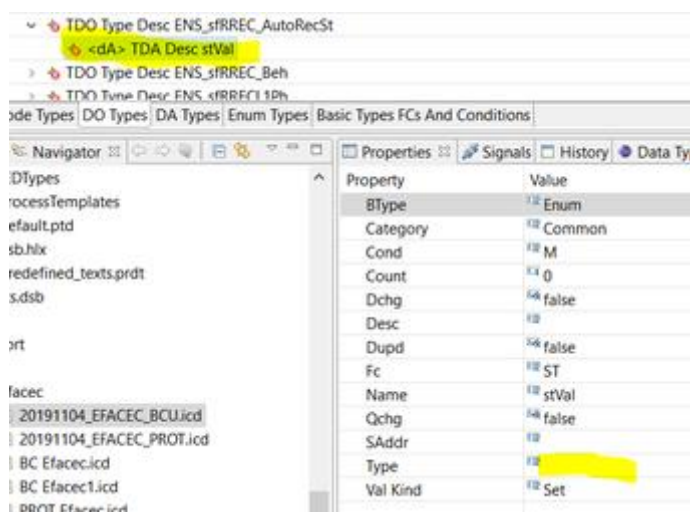


Figure 9 - Function template validation example

3.3.1.3 Assign Namespace

It is recommended that LNodeTypes used in function templates contain the reference of the used datamodel namespace.

If the function template has the namespace in the LN, use that one; otherwise, ask what name space to be used; allow granularity of selecting a namespace per function.

Property	Value
Application	EntsoE-Integrated Specification Tool
Desc	
Name	Entso-E-IST
Revision	B
Version	2007

Figure 10 - Namespace selection example

NameSpaces for Logical Nodes are assigned by using the proper LNode Type.

Additional checks:

- Check if all DO/DAs are present in the Masternamespace
- Fix incoherent DataTypes from Masternamespace. Selection of Enums etc.

It is recommended for tools to upgrade all datamodel namespaces to the Masternamespace of the SSD file.

The level of completion (for example adding of substitution, choice of ENUM values) shall be defined by the user.

Example of a fix by adding missing Enumerations:

Property	Value
BType	Enum
Category	Common
Cond	M
Count	0
Dchg	false
Desc	
Dupd	false
Fc	ST
Name	stVal
Qchg	false
SAddr	
Type	AutoRecSt
Val Kind	Set

Figure 11 – Enumerated element missing different values

▼ ◆ TEnum Type Desc AutoRecSt
◆ TEnum Val Aborted
◆ TEnum Val CBclosed
◆ TEnum Val CycleUnsuccessful
◆ TEnum Val FaultDisappeared
◆ TEnum Val InProgress
◆ TEnum Val Not ready
◆ TEnum Val Ready
◆ TEnum Val Successful
◆ TEnum Val TripFromProtection
◆ TEnum Val Unsuccessful
◆ TEnum Val WaitingForTrip
◆ TEnum Val WaitToComplete

Figure 12 – Enumerated values added by SST

3.3.2 Documenting a single line in the .ssd file

The user has the possibility to document the single line by adding conducting equipment elements to the substation section of the .ssd file



Figure 13 - Osmose single line example

3.3.3 Instantiating functions and subfunctions

When instantiating functions and subfunctions in a specification .ssd, these elements are converted from a template to a unique instance. This means a unique naming for given SCL parent is required for each instance.

Should the function template contain additional private elements, then these should be copied as well¹¹.

¹¹ This is a recommendation for SST tools

Functions may be inserted at different levels of the specification (substation, voltage, bay) depending on the application.

In case multiple instances of the same (sub)function are inserted, the SST is recommended to handle the uniqueness of the specified datamodel in an automated way. If this is not possible the user will need to manually fix this.

3.3.3.1 Example of an instantiated function:

The Function ProtCBI is a collection of Sub Functions with different XCBR¹² LNodes for the different kinds of Switches in a bay

```
<Function name="ProtCBI">
  <Private type="eIEC61850-6-100">
    <SubFunction name="CBIphsA">
      <LNode IedName="SAF1Q1PROT" IdInst="ProtCBI" LnClass="XCBR" LnInst="3" LnType="XCBR_CBIphsA" prefix="CBIphsA">
        <Private type="eIEC61850-6-100">
          <eIEC61850-6-100:LNNodeInputs>
            <eIEC61850-6-100:SourceRef input="Trip phase A"/>
            <eIEC61850-6-100:SourceRef input="General trip 3Ph"/>
          </eIEC61850-6-100:LNNodeInputs>
          <eIEC61850-6-100:LNNodeSpecNaming sIedName="SAF1Q1PROT" sIdInst="ProtCBI" sLnClass="XCBR" sLnInst="3" sPrefix="CBIphsA"/>
        </Private>
      </LNode>
    </SubFunction>
    <SubFunction name="CBIphsB">
    <SubFunction name="CBIphsC">
  </Private>
</Function>
```

Figure 14 - Function example with inputs

As shown, these inputs are open. The SourceRef just contains the input variable attribute, but no pointer to the incoming data. In the finalized specification or configuration, these should be completed for correct dataflow documentation.

The specification attributes in the LNNodeSpecNaming element of the LNode are initialized with the same values as the main LNode attributes. After implementing this LNode with an LNode from an IED they will keep their original specified values.

In order to guarantee the uniqueness of the LNode datamodel, tools may modify the prefix or suffix¹³.

3.3.3.2 Introduction of the PowerSystemResource Reference

(Sub)functions may be linked to primary equipment by

- Placing them hierarchically under the conducting equipment as an EquipmentFunction

¹² XCBR is the standardized IEC61850-7-4 LN type for circuit breaker interface

¹³ It is recommended for the user to define the rules and for SST tools to implement them in a way that rules are applied to the datamodel in an automated way.

- Adding a PowerSystemResource¹⁴ element to the subfunction, pointing to the primary equipment. This solution adds more flexibility and allows linking a subfunction to multiple conducting equipment, or to separate functional hierarchy from primary device hierarchy.

Example:

In order to unambiguously assign a CSWI¹⁵ to the switch it is intended to control, we need to add a Power System Resource Reference for each CSWI.

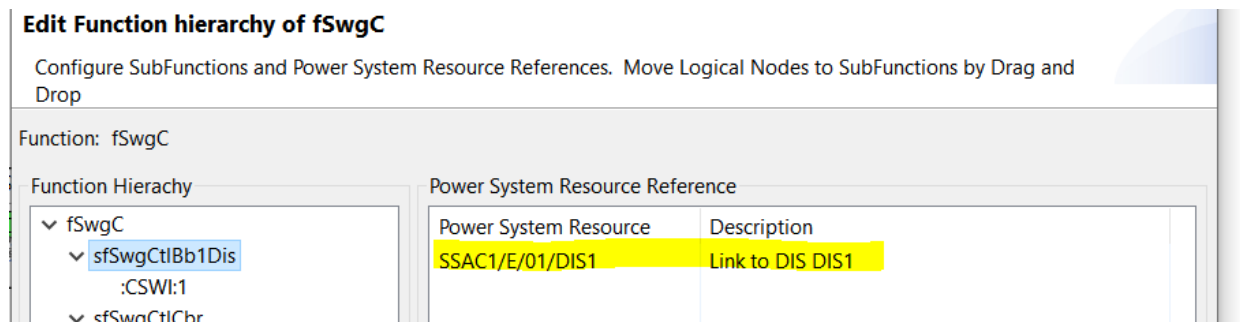


Figure 15 - Power System Resource example

```
<Substation sxy:y="9" name="SA">
  <VoltageLevel name="F1">
    <Voltage multiplier="k" unit="V">110</Voltage>
    <Bay name="Q1">
      <ConductingEquipment name="QA1" type="CBR"/>
      <ConnectivityNode name="grounded" pathName="SA/F1/Q1/grounded"/>
      <Function name="fCBIf">
        <Function name="ProtCBI">
          <Private type="eIEC61850-6-100">
            <eIEC61850-6-100:PowerSystemResources>
              <eIEC61850-6-100:PowerSystemResource resource="SA/F1/Q1/QA1"/>
            </eIEC61850-6-100:PowerSystemResources>
          </Private>
          <SubFunction name="CBIPhsA">
            <SubFunction name="CBIPhsB">
              <SubFunction name="CBIPhsC">
                </Function>
              </SubFunction>
            </SubFunction>
          </SubFunction>
        </Function>
      </Function>
    </Bay>
  </VoltageLevel>
</Substation>
```

Figure 16 - PowerSystemResource SCL example

3.3.4 Creating dataflow between subfunctions

Dataflow links between subfunctions shall be defined as Soureref or Controlref elements (elements documented in 10.3 Annex C: Enhancements of SCL for virtual signal flow and virtual IED (ID7.3.C)). Each SourceRef element represents an input coming from another subfunction. A ControlRef is placed at a controlling subfunction, and will point to the

¹⁴ It is up to WG10 to define for what and how the PowerSystemResource element is to be used

¹⁵ CSWI is the standardized IEC61850-7-4 LN type for circuit breaker or switch controlling

subfunction element that is controlled. If a SourceRef or ControlRef placeholder already exists in the function template, this may be multiplied during specification stage if multiple subfunctions need to connect to this reference.

Sources of SourceRefs point to a DO or DA. It is recommended for WG10 to define how during implementation IED dataflow will be generated. For example: if a source points to an stVal DA, real IED dataflow will probably also need to contain q quality and t timestamp. For Osmose, each DA was created separately as a SourceRef in the substation section, in order to facilitate dataflow implementation.

The SourceRef contains an optional “service” attribute, which allows to specify the expected communication service. Examples are

- GOOSE
- MMS reporting
- Internal
- Wired
- Poll
- SMV

The communication service type element may be used to automate the creation of dataflow implementation in later engineering process stages. It is recommended for users to also specify dataflow which is considered internal (in the IED), because it allows to fully document system behavior and provides flexibility in case an internal specified data exchange needs to be implemented between 2 IEDs.

The function template already contains the inputs for each (sub)function, but during specification stage the user might need to modify these (adding/removing) linked to the application. It is required for SST tools to enable this possibility.

The “input” attribute of the SourceRef is considered as the unique identifier of the input. Within Osmose its content will be identical to the “desc” attribute. Linking the inputs attribute with a DO/DA description may automate dataflow creation between subfunctions, but some use cases are to be considered:

- A subfunction input may require duplication if coming from multiple other subfunction outputs
- A “start autorecloser” input coming from a “protection function operate”, creating a mismatch between input and DO/DA description

It is recommended for WG10 to define rules regarding the use and content of the input attribute

- For automation of dataflow between (sub)functions
- For implementation in real IEDs

Within the SourceRef element an extRefAddr attribute is used to point to the ExtRef in the implemented LN of the real IED.

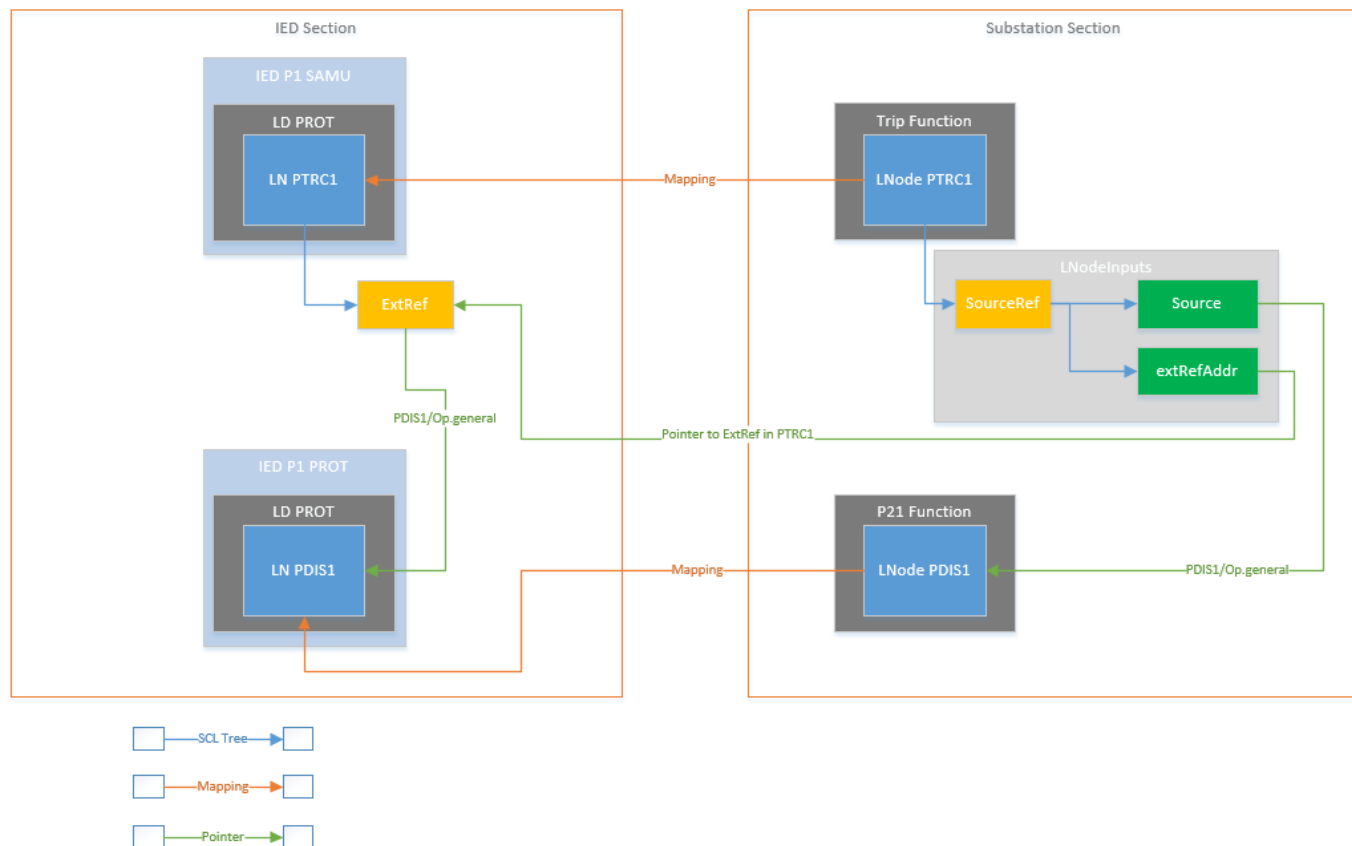


Figure 17 - Example of SCL specification and implementation of extRefAddr

In a specific use case this element allows to document a deviation between the LN implementing the LNode requirement and the LN containing the ExtRef with the received data. For example: A PDIS LNode is mapped to a PDIS LN, but the GOOSE receiving LN is a generic GGIO.

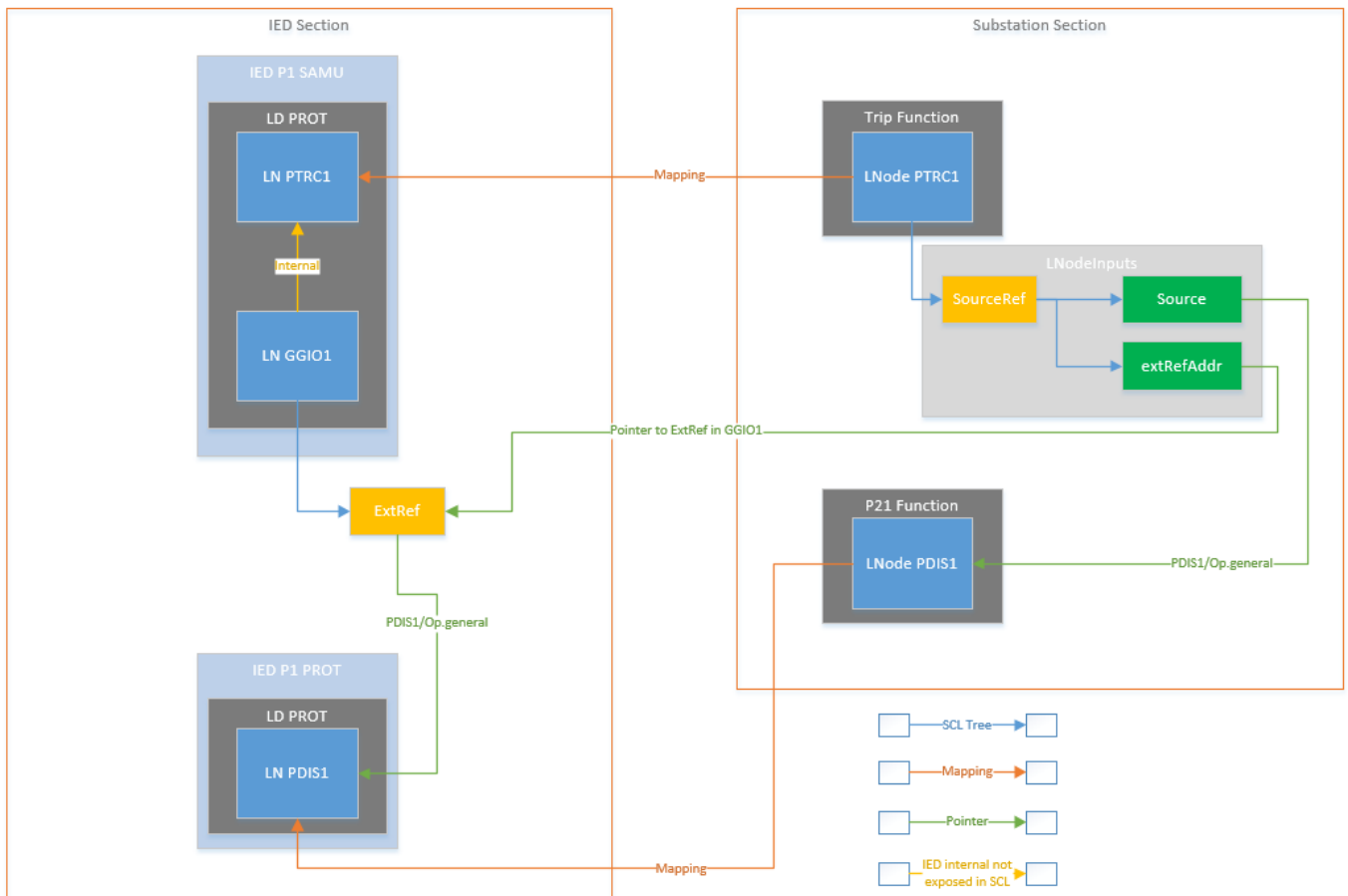


Figure 18 – Example of extRefAddr allowing documentation of deviation in receiving LN and function LN

This modelling can also be applied for MMS reporting dataflow, to document a difference between the function LN (for example a proxy LN) and a client LN (for example an ITCI instance).

Regarding the extRefAddr semantic, it is recommended for WG10 to define semantic rules for this attribute to cover different scenarios

- A 1 on 1 relationship between extRefAddr of a SourceRef and an ExtRef implemented in the real IED
- An extRefAddr of a DO SourceRef with multiple DA ExtRef implemented in the real IED
 - o Example: 1 DO SourceRef Pos implemented by 3 DA ExtRefs stVal, q and t
- An extRefAddr of multiple DA SourceRefs with 1 DO ExtRef implemented in the real IED
 - o Example: 3 DA SourceRefs stVal, q and t implemented by 1 DO ExtRef Pos

A possible solution may be to allow the ICT and SCT to split DO SourceRefs into different DA SourceRefs to create coherent combinations of extRefAddr and ExtRef elements.

The source element of a SourceRef may point to a DO or a DA. For facilitation of the automated creation of dataflow, Osmose implemented for GOOSEs separate SourceRefs for each DA required in the dataset (for example stVal and q). If the source element points to a DO, the SCT requires additional rules to create datasets in the real IEDs. It is recommended for WG10 to investigate the use of a new element “servicespec” inside the SourceRef to indicate for the SCT how dataflow needs to be implemented, in case a source points to a DO.

3.3.5 Allocation of subfunctions to virtual IEDs

As described in the introduction, a user may want to specify the physical allocation of functions to IEDs and describe the dataflow requirements for the physical IEDs. In order to allow this, the specification .ssd will contain virtual IEDs which may be exported from the .ssd as an .isd file (IED specification description).

The virtual IED datamodel will be created from the content of the LNode linked to each subfunction that is mapped to the IED by the user. If the LNode elements are not sufficient for the creation of the virtual IED datamodel (for example because of missing LdInst or prefix), the SST shall:

- Autogenerate the virtual IED datamodel structure based on predefined rules¹⁶
- Request the user to specify the missing elements in order to create the virtual IED datamodel
- Additional elements can be used to specify specific IED requirements
 - The communication section for requirements related to physical communication interfaces
 - IEC61850-90-4 logical nodes for requirements related to communication channels
 - Logical nodes like LPHD¹⁷ for additional physical requirements

When creating virtual IEDs, preconfigured values from the substation section will be taken over into the preconfigured values in the IED section.

With the use of a virtual IED, the user can specify additional requirements like e.g. grouping for setting groups, specific communication requirements, etc. For this the service section of the IED will be used¹⁸.

As part of the function allocation, it may as well be required that the SST needs to be able to duplicate a function (e.g. the breaker interface function if both protection and control have a wired interface to the breaker).

¹⁶ It is recommended for WG10 to specify and standardize these rules

¹⁷ LPHD is the standardized IEC61850-7-4 LN type for physical device interface

¹⁸ It is recommended for WG10 to enhance the service section so that the user can maximally specify its requirement (for example specify that all datasets and report control blocks need to be in LD0)

Depending on the use case the virtual IED may already contain dataflow elements

- Maximally configured in case of a project
 - o It is recommended for tools to include communication supervision in an automated way based on the services section of the virtual IED
- Minimally in case of a template

Remark: If the user wants to specify function allocation constraints, he is forced to introduce virtual IEDs, allocating a (sub)function to a single IED. It is recommended to WG10 to introduce a solution for this use case which allows to introduce allocation constraints and support more complex variant scenarios like for example:

- Allocating a function to P1 main protection in case the voltage level is 70kV
- Allocating a function to P2 backup protection in case the voltage level is 150kV

3.3.5.1 Creating dataflow between virtual IEDs

A user has the option to create the dataflow between virtual IEDs at specification stage. This will be done by the SST based on the dataflow between subfunctions allocated to the different virtual IEDs and taking into account specifications from the services from the virtual IEDs.

3.3.6 Finalized ISD/SSD files

Depending on the use case, the finalized SSD and ISD files are considered as project instances or templates. It is recommended for WG10 to clarify how the SCL content of these files may reflect their use (instance or template) in a clear way, understandable by tools. This notion is important because it defines what return is expected from the next steps of the engineering process.

Chapter 4 is providing an overview of the required SCL content of an ISD file.

3.4 IED selection and Procurement

Once the specification is finalized, the user wants to select the appropriate IEDs for implementation. Based on the SCL files the user uses for selection an SCT can propose a best fitting IED. Parameters of IEDs that are not exposed in SCL (price, supported environmental conditions, size etc) can also be taken into consideration, but

- It is up to the user to define the priority of the additional parameters/requirements
- An SCT may optionally take these additional information account for supporting the user in the decision-making process, but outside the scope of the SCL file content

A user wants to select IEDs for different use cases:

- IED instances for a specific project
- IED templates to populate a library, and to use in projects

3.4.1 IED selection

Once the specification is ready, IEDs need to be selected for implementation of the specification. In order to evaluate and select based on the SCL files, there are 2 variants possible

- Selection from an existing available ICD library
- Selection and procurement of vendor IEDs

The different possible options are described below

Process step input (output from the specification step):

- A. ISD (series)
- B. SSD without virtual IED

Selection from:

- A. Library of extended ICD files with a populated substation section (from other specification files)¹⁹
- B. Library of ICD files without a populated substation section (vendor library)²⁰
- C. Specific offer from vendor based on
 - a. ISD (offer = a single ICD)
 - b. SSD (offer = series of ICDs)

Output:

- A. A series of ICDs to cover a series of ISDs (1 to 1 relationship between ICD and ISD)
- B. A series of ICDs to cover an SSD file without virtual IEDs
- C. A series of ICDs to cover a single ISD (if no exact match can be found)

All output SCL files will contain a substation section that is engineered to respond to the files stated as input for this process step.

The different scenarios are detailed in chapters below. The content provides recommendations for the SCT (substation configuration tool) that will host the required

¹⁹ Not part of Osmose project but added for completion of possible scenarios

²⁰ Not part of Osmose project but added for completion of possible scenarios

functionalities in order to help the user select the appropriate IEDs, as well as for the ICT responsible for offering one or more ICDs for covering a specification.

In order to evaluate if the selection corresponds to the specification, a comparison between the specification and the selected/offered IED is required. Details about comparison between SCL files are given in chapter 0.

3.4.2 Creating an extended ICD with enhanced substation section

In use cases where the user is expecting a specific IED offer to its specification, ISD or SSD files exposing the system specifications from the user need to be answered by a manufacturer or integrator with (a selection of) IEDs. The ICD that is offered is considered to contain elements from the specification substation section with mapping references to the offered IED datamodel. This will offer

- The user the possibility to verify how and to which level the specification is implemented by the IED
- The IED supplier the possibility to document how the specification will be implemented by the IED

Details about the exact SCL content of the extended ICD file are provided in 4.1 ISD/ICD content. The substation section contains the necessary elements to allow a traceability of the specified vs the implemented datamodel.

The ICT is considered to produce this extended ICD file, based on IED capabilities, and the provided specification files (SSD/ISD).

In case a physical allocation of functions is requested by the user, it is recommended for users to provide ISD files to IED manufacturers, in order to

- limit the work to be done by the manufacturer
- Limit the risk of getting an offer which does not fully cover the expectations

In case of an SSD without allocation of functions to virtual IEDs is given to the IED supplier, the IED supplier is considered to have complete freedom to decide which functions to allocate to which device.

A detailed overview of the steps to be executed in the ICT is available in 10.1 A: Interoperability test specification (ID7.2.A) – test cases 151 and 152

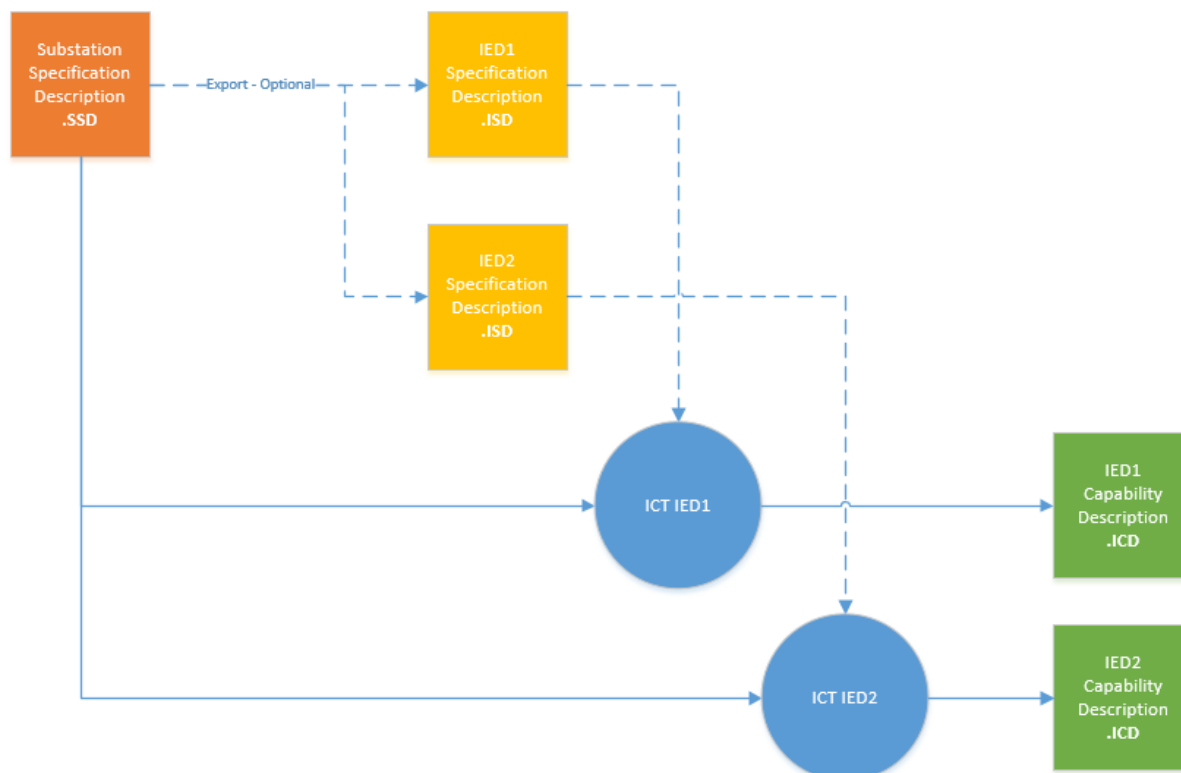


Figure 19 - ICD creation

3.4.3 Selection of IEDs based on a specification

In order to select IEDs corresponding to the specification needs, a process step is required to compare the specification with the IED offer (from a library of specific offer). It is considered that this comparison and selection step is executed at the beginning of the configuration stage which will deliver a project or template system configuration (SCD). It is a recommendation for the SCT responsible for this configuration to implement the functionalities as described in the following chapters.

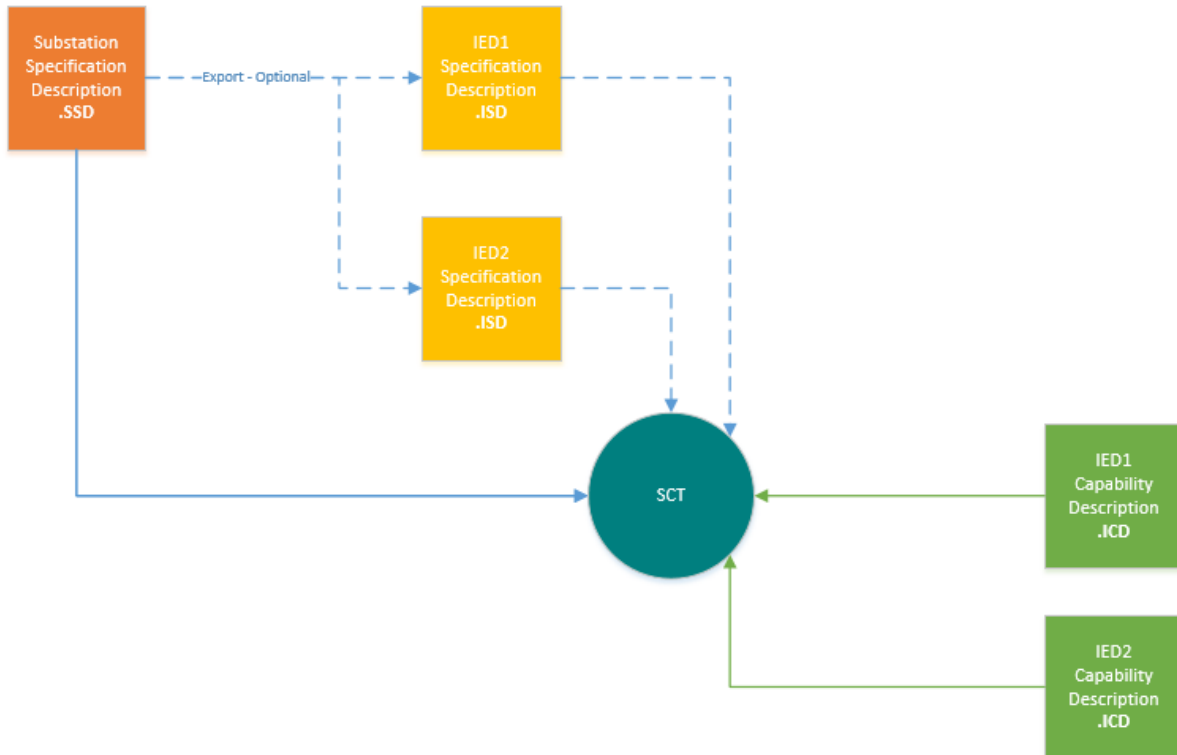


Figure 20 - Selection of IEDs in the SCT

3.4.3.1 Selection of IEDs based on an ISD input

3.4.3.1.1 From an existing IED (ICD) library

3.4.3.1.1.1 With a populated substation section

The substation section of the ICD is populated from the input of another specification and may not correspond to the input specification. In order to select the best matching IED, the SCT will search for (in a descending order – if not found in first step, move to second step)

1. In the ICD substation section for a corresponding LNode
 - a. Maximal full specification model (including subfunction name and type) up to DO/DA part of the substation section dataflow
 - b. Minimal LNTYPE (including required DO/DA for dataflow)
 - i. Optionally including additional DO/DA not present in dataflow but part of the specified DataTypeTemplate section
2. In the ICD IED section for a corresponding logical node
 - a. Maximal full LNodeSpecNaming up to DO/DA part of the substation section dataflow
 - b. Minimal LNTYPE (including required DO/DA for dataflow)

- i. Optionally including additional DO/DA not present in dataflow but part of the specified DataTypeTemplate section

3.4.3.1.2 Without populated substation section

The substation section of the ICD is empty, in order to identify if an ICD matches the requirements, the SCT will search in the IED datamodel for each subfunction LNode a corresponding logical node

- a. Minimal LNTYPE (including required DO/DA for dataflow)
- b. Maximal full LNTYPE compatibility (including all DO/DA of the specification LNTYPE)

3.4.3.1.3 From a specific vendor offer

In this variant the substation section is populated with content from the substation section from the input files, and guarantees a 1 on 1 relationship between the input ISD and the offered ICD. In this case it can be considered that the function/subfunction structure from the ISD is completely retained in the offered ICD file. In order to correctly respond to the ISD requirement, the vendor should offer a completely conform ICD.

When comparing the offers from different vendors, selection by the SCT of the best-suited offer will be based on (in descending order)

1. Mapping fully corresponding to specified datamodel
2. Mapping corresponding to specified LNclass including DO/DA
3. Mapping corresponding to specified LNclass with DO/DA deviated mapping
4. Mapping of a specified LNclass with another LNclass LNode with mapping of DO/DA

3.4.3.1.4 Other ICD evaluation criteria

Additionally, a comparison between specified and offered service section is required to verify if the required services are supported. Especially for IEDs not supporting later binding a services comparison is necessary to evaluate if the required dataflow can be implemented. It is recommended to WG10 to specify the minimal content of the ISD services section, in case dataflow is configured.

Regarding mapping of dataflow, evaluation is required of the level of mapping of SourceRefs to Extrefs in the offered IEDs. The Extrefaddr attribute of the Sourceref in the substation section can be used to evaluate this (for IEDs which support later binding).

3.4.3.2 Selection of IEDs based on an SSD input

In this case it is considered that no physical allocation requirements are given by the user, and that the goal of the selection is to cover the requirements from the full SSD with a series of physical IEDs. By comparing SCL content, an SCT tool may propose to the vendor different combinations of series of IEDs covering the full specification, but it is up to the user to decide which combination fits best to his needs.

3.4.3.2.1 From an existing IED (ICD) library

Identical selection criteria are applied as stated in 3.4.3.1.1, with the added complexity that an optimization is required to provide the best matching set of ICDs. In order to provide directions to the SCT selection logic, the user may define priority settings like

- Minimal number of IEDs
- Maximal correspondence to the specified datamodel
 - o In substation section (subfunction definition and LNode datamodel)
 - o In IED section (datamodel)

3.4.3.2.2 From a specific vendor offer

In this case the vendor has optimized his offer for covering the full specification .ssd file with a series of IEDs, offering a series of .icd files. The same selection evaluation is applied as in 3.4.3.1.3, but now taking into account the group of proposed .icd files.

3.5 Configuration

The configuration step includes the creation of a system configuration based on the inputs from previous process steps.

A detailed overview of the steps to be executed in this phase is available in 10.1 A: Interoperability test specification (ID7.2.A) – test cases 211 and 212

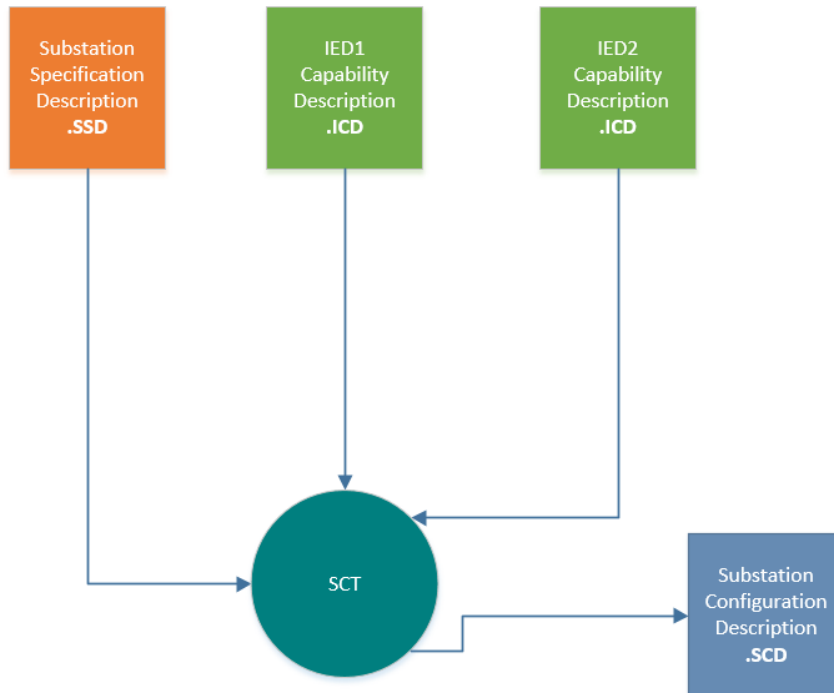


Figure 21 - creating the SCD file

In this stage the SSD file is the starting point:

- Without virtual IEDs:
 - o importing the IEDs from the .ICD files and configuring dataflow between them
 - o updating the substation section with the mapping from the ICD files
- With virtual IEDs:
 - o Replacing the virtual IEDs from the SSD file with the real ones from the ICD files
 - o Implementing the configured dataflow between virtual IEDs in the real IEDs

Dataflow between IEDs is configured as described by IEC61850-6, with links to the substation section by configuring extRefAddr attributes in the SourceRef element of the LNodeinput. If IEDs support later binding these links are already preconfigured in the ICD files. Please see 3.3.4 Creating dataflow between subfunctions for more details about the use of extRefAddr.

It is recommended for IEDs to support creation of ExtRefs by the SCT at any LN of the IED, in this way facilitating the dataflow implementation by the SCT.

If creating dataflow between real IEDs consists of the creation of control blocks and datasets, it is recommended for configuration tools to allow users to set rules for dataflow implementation. These rules may contain for example:

- Dataset and control block creation and attributes (amount, naming, optional fields etc)
- Location of datasets and control blocks in the IED

Remark: the SCD configuration contains the configured dataflow. The interpretation of incoming data and behaviour of the IEDs is a local issue. It is recommended for WG10 to provide ways to document logical behaviour at specification level for the purpose of

- System testing and simulation
- System behaviour documentation
- Enabling of logical behaviour configuration automation in IEDs

3.6 Commissioning

When the SCD file is ready, configuration needs to be loaded in the IEDs which makes the system ready to be tested.

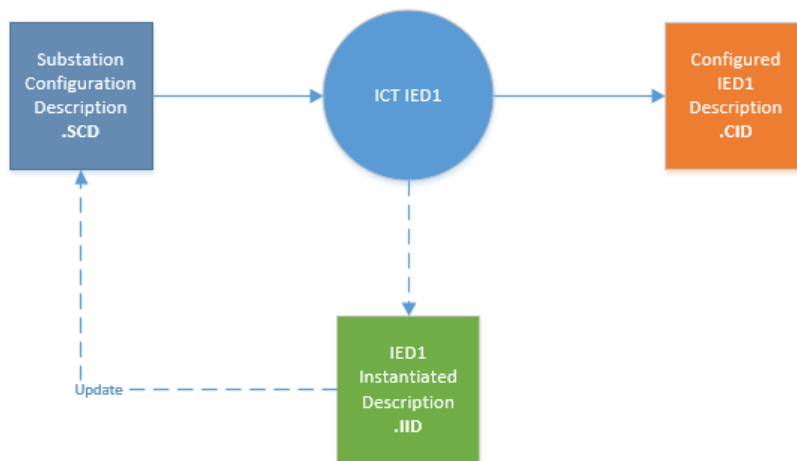


Figure 22 - Creating IED configuration

The process, as defined in IEC61850-6 shall be executed.

Remark: if additional SCL configuration of the IED is added in the ICT (for example related to communication supervision), the SCD file needs to be updated with the new version of the IID file. It is recommended for IED manufacturers to allow a full SCL configuration of the IED in the SCT to avoid this additional step and raise the overall process efficiency.

3.7 User right management during the engineering process

It cannot be considered that during the execution of the engineering process the same person manages and handles the SCL files related to specification and configuration. We can differentiate multiple scenarios where (a part of) SCL files are handed over to another person for further engineering process steps, optionally with the expectation of receiving it back in a later stage.

- Between specification and configuration: a specification engineer hands over a specific bay for configuration to a configuration engineer
- Between configuration and commissioning: a configuration engineer hands over a configured bay for execution and testing to a commissioning engineer, and allows him to correct errors during testing.
- Between different substations: engineers from different companies exchange specification files of different substation to configure inter-substation dataflow (please see chapter 0 for more details).
- During configuration: The user hands over the system configuration to the busbar protection integrator to integrate the busbar protection.

It is recommended for WG10 to provide possibilities to support the above use cases with a clear and efficient version and rights management.

4 SCL file content

4.1 ISD/ICD content

Below a table is shown providing the detailed content of each filetype. Template and project usecases are covered.

		isd	icd based on isd file	icd based on ssd file without virtual IEDs
Substation section	Substation name	as defined by user in export (TEMPLATE or project specific depending on use case)	Same as isd	Same as ssd
	Voltage name	as defined by user in export (TEMPLATE or project specific depending on use case)	Same as isd	Same as ssd
	Bay name	as defined by user in export (TEMPLATE or project specific depending on use case)	Same as isd	Same as ssd
	Other voltages/bays	If applicable to the IED	If applicable to the IED	If applicable to the IED
	conducting equipment	Optional (mandatory if LNode below is kept or referenced via	Optional (mandatory if LNode below is kept or referenced	Optional (mandatory if LNode below is kept or referenced

		powersystemresource)	via powersystemresource)	via powersystemresource)
	Functions / subfunctions	Optional (mandatory if LNode is present)	Optional (mandatory if LNode is present)	Optional (mandatory if LNode is present)
	LNode	LNodes mapped to the IED and (optionally) LNodes providing dataflow to the specified IED remove any other LNodes (for simplicity)	IED mapping LNodes only (from real IED) Remark: LNodes may be added in this stage that were not part of the isd, to guarantee a correct implementation of the required behavior (LNodespecnaming shall remain empty in this case)	IED mapping LNodes only (from real IED) Remark: LNodes may be added in this stage that were not part of the ssd, to guarantee a correct implementation of the required behavior (LNodespecnaming shall remain empty in this case)
	LNtype	yes	LNtypes from specification	LNtypes from specification
	DOS/DAS for mapping documentation	NA	To be added to document DO/DA mappings (only if not exactly identical to specification)(to use to document if LNode is mapped to multiple logical nodes)	To be added to document DO/DA mappings (only if not exactly identical to specification)(to use to document if LNode is mapped to multiple logical nodes)
	DOS/DAS with specified values	yes - if specified by user (only for own IED)	yes same as ISD - for representation of physical description elements (LPxx) of an IED which does not support physical interface LNs, it is allowed to add its physical description as DOS/DAS. DOS/DAS is also used to document datamodel deviations (see Annex C: Enhancements of SCL for virtual signal flow and	yes same as SSD - for representation of physical description elements (LPxx) of an IED which does not support physical interface LNs, it is allowed to add its physical description as DOS/DAS. DOS/DAS is also used to document datamodel deviations (see Annex C: Enhancements of SCL for virtual signal flow and

			virtual IED (ID7.3.C)	virtual IED (ID7.3.C)
	LNodeSpecNaming (s-attributes)	yes (with same content as LNode)(created/exported by SST)	yes - recovered from ISD (original ISD IED name)(mandatory element in ISD)	yes - recovered from SSD (mandatory element in SSD)
	LNodeinputs: SourceRefs²¹ inputs on LNodes	yes - without source if LNode is not present in the substation section	yes, from isd - without source if LNode is not present in the substation section	yes, from ssd - without source if LNode is not present in the substation section
Communication section (IP-adress, subnet mask, gateway)		Yes - with predefined values if applicable for the usecase	yes, template values (from original icd) with optionally additional information from isd	yes, template values (from original icd)
IEDSection	IED name	as defined by user in export (TEMPLATE or project specific)	TEMPLATE	TEMPLATE
	Service section	Optional - to specify IED requirements	Yes - with real IED information	Yes - with real IED information
	Datasets and control blocks creation	Yes optionally, if you want to specify dataflow on IED level. No if you use isd file only to physically allocate (sub)functions to devices and specify a template IED	Optional - if dataflow configured in isd	Optional
	DOI/DAI with values	Yes - if specified in substation section	Only if supported by the IED - values from isd file IED section	Only if supported by the IED - values from DOS/DAS from ssd file

²¹ Source elements always point to the specified LNodes, also in configuration stage

	Extref	Optional and detailed content depending on the use case (template or project) -If used: With minimally desc and servicetype and maximally all later binding elements -For dataflow between substations: An Extref with at least p-attributes is required (when IED from other substation is not specified). If the IED from the other substation is fully configured (including communication (control blocks etc)) the Extref can be fully populated	Default behavior: same content as isd file Exception: Mandatory in case an LNode is modelled in the real IED by another logical node (documented via ExtRefAddr of LNodeinput)	Optional: Extrefts for later binding of real IED dataflow Exception: Mandatory in case an LNode is modelled in the real IED by another logical node (documented via ExtRefAddr of LNodeinput)
DataTypeTemplates	LNtypes	LNtypes from substation section and virtual IED (in specification mandatory DO/DA are not required)	LNtypes from real IED (from IED section) and the original ones from the substation section - optionally extended with mappedDoName and mappedDaName (mandatory DO/DA are required)	LNtypes from real IED (from IED section) and the original ones from the substation section - optionally extended with mappedDoName and mappedDaName (mandatory DO/DA are required)
	Values	if specified only	Point to be addressed by WG10 For Osmose: values only copied if LNtype (specification) is copied to icd file	Point to be addressed by WG10 For Osmose: values only copied if LNtype (specification) is copied to icd file

5 Comparison between SCL files

During the execution of the engineering process, some steps require comparison of SCL files in for evaluation of:

- A received IED offer (ICD vs ISD/SSD)
- The implementation of the specification in the configuration (SCD vs SSD)

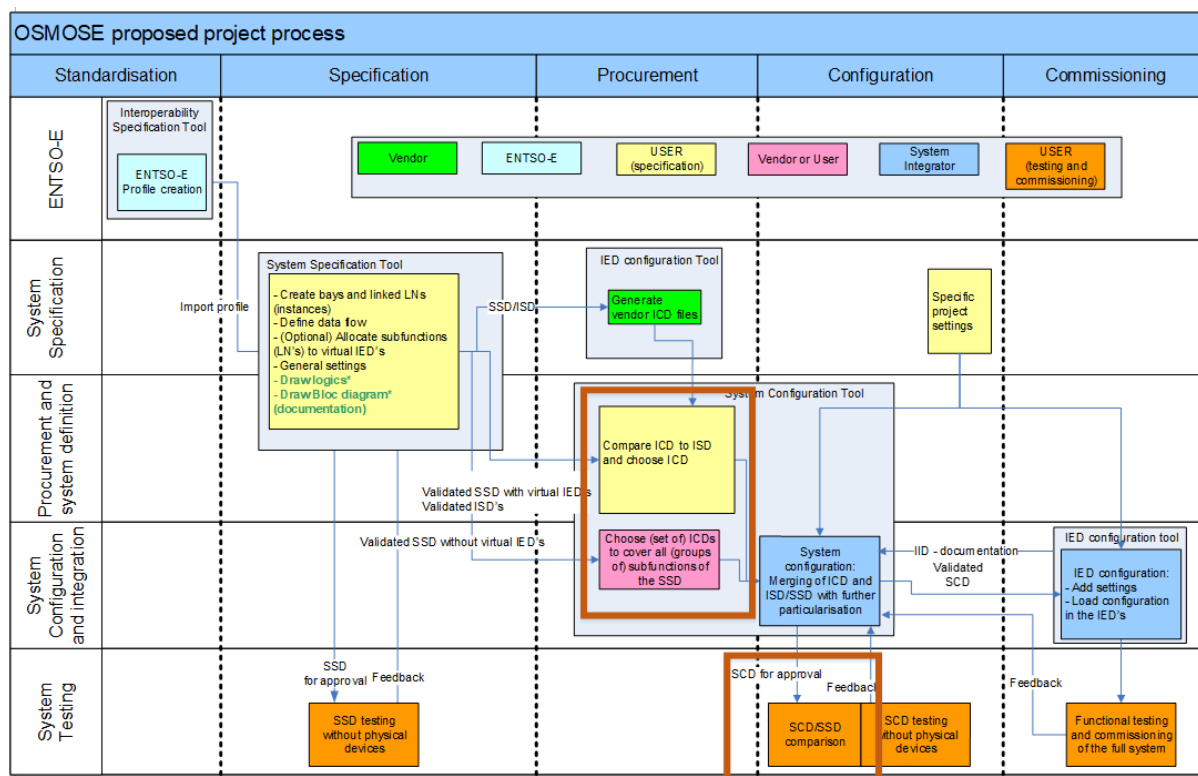


Figure 23 - Comparison process steps

It is recommended for the market to provide tools allowing these comparisons between SCL files. A detailed specification of all use cases and requirements is available in Annex B.

Additional SCL file comparisons are useful, related to documentation

- Between an implemented instance of a template and the template definition (for example an .fsd template versus instance).
- Between different versions of the same file to document changes.

This type of comparison is not further investigated by Osmose, because not directly related to the efficiency of the engineering process. It is nevertheless recommended for all documentation tools to include these kinds of comparison functionalities.

6 Engineering process for dataflow between 2 substations

In some use cases substations may need to exchange data between each other for protection and control purposes (for example for teleprotection purposes). In this case IEDs from a substation will need to subscribe to information from IEDs of another substation. This chapter describes the specificities of these specification and configuration steps.

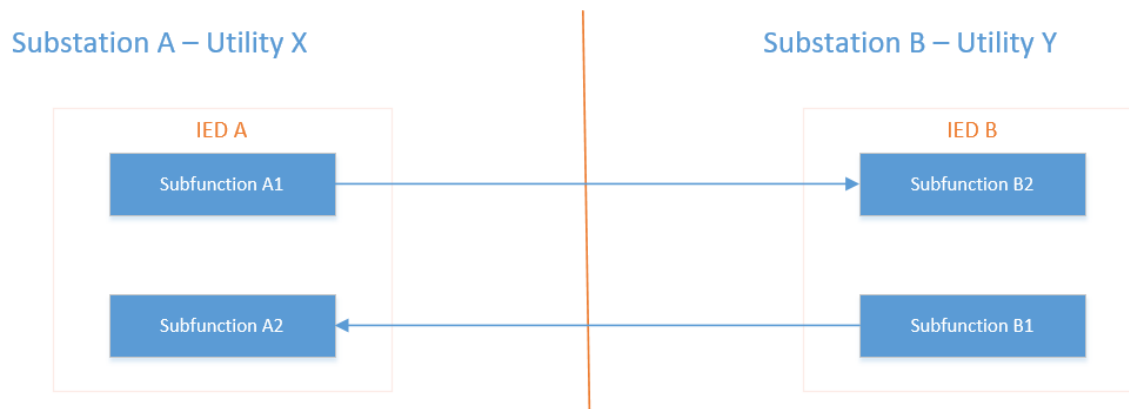


Figure 24 - Usecase for intersubstation dataflow

At the beginning of this process, it is considered that engineers from both substations have agreed on the data that is to be exchanged, and on the datamodel that is used. The process starts considering all engineers have this knowledge.

We will differentiate flows with or without use of virtual IEDs in the SSD

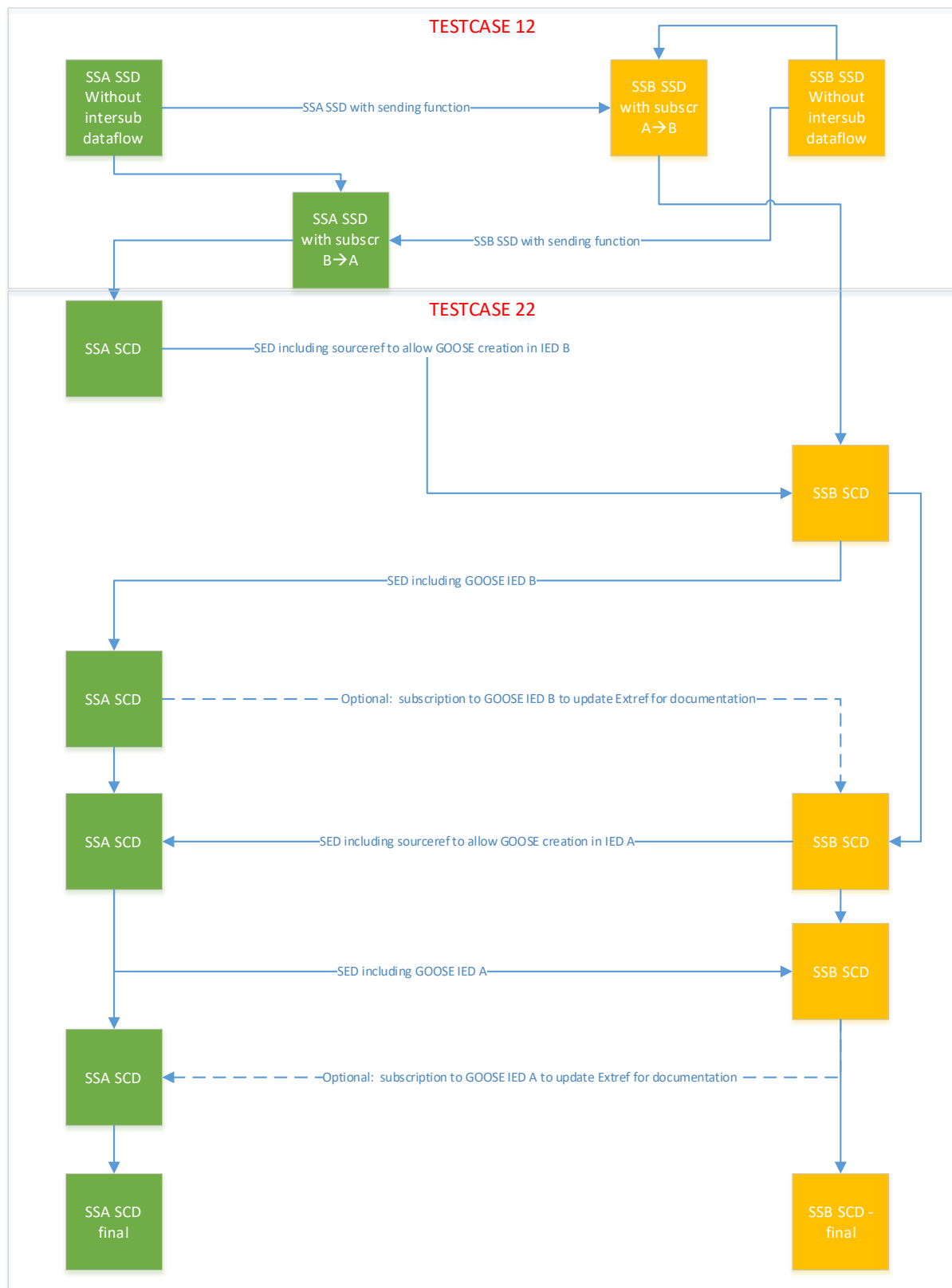


Figure 25 – SSD/SED exchange process without virtual IEDs

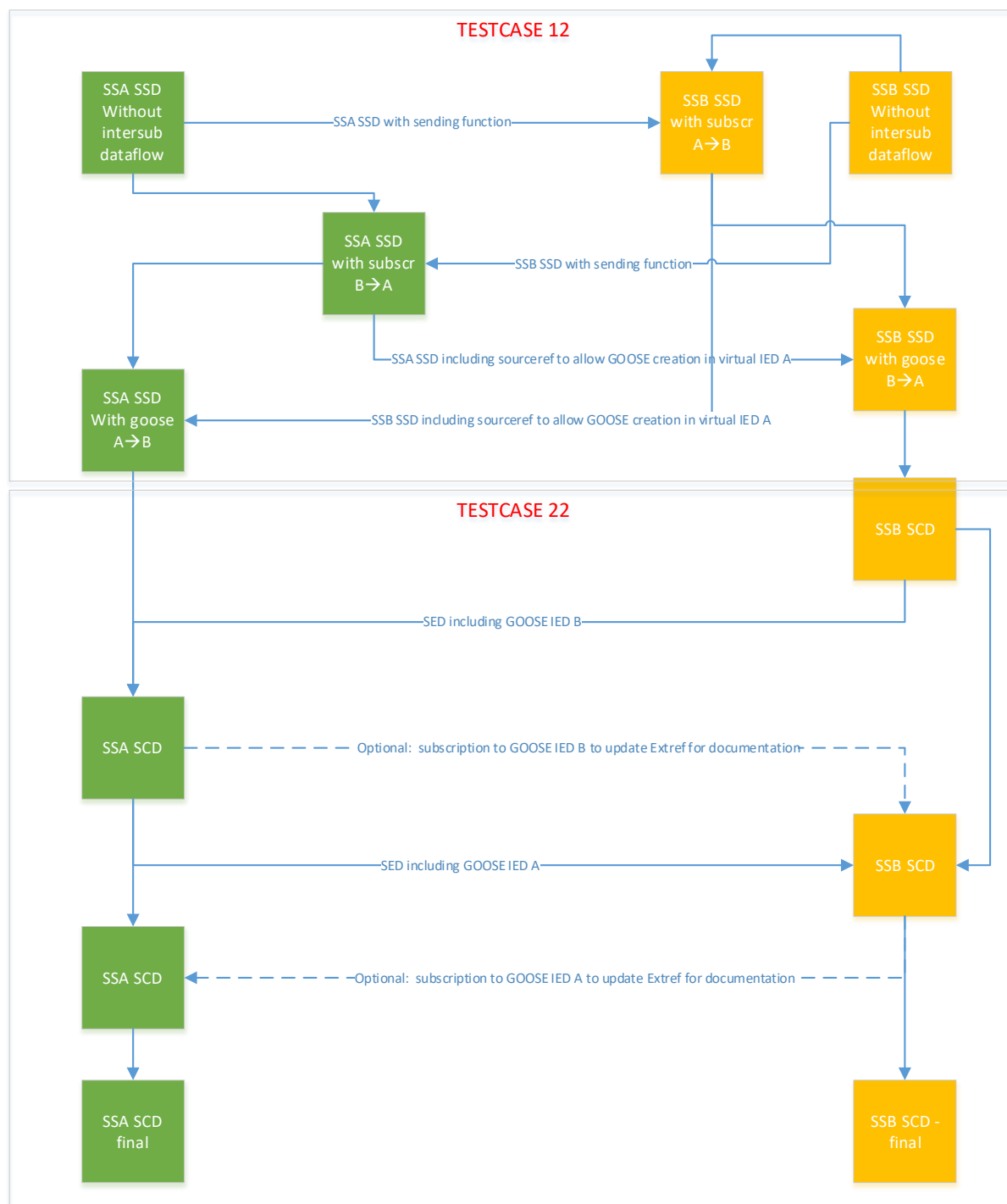


Figure 26 – SSD/SED exchange process with virtual IEDs

Description of each test case step is covered by Annex A: step 12 and step 22, as indicated on the above figures.

It is recommended for WG10 to define the required content of the substation section of the exchanged SCL files between SSA and SSB, related to the exchange of SSD files during specification stage. A recommendation from Osmose (and executed within Osmose) is to

limit the file content to the exchanged information only, keeping in the substation section only the functions and LNodes which exchange information.

7 Documentation

The additional elements of the substation section related to specification and implementation require a clear documentation for users during commissioning and maintenance.

- Functions and subfunctions (optionally linked to primary conducting equipments)
- Dataflow between functions and subfunctions
- Implementation of the specification by IEDs
 - o Dataflow
 - o Function/subfunction allocation

It is a recommendation for engineering tools (specification and configuration) to implement documentation generation based on the SCL content.

8 Testing and Simulation

Concepts developed in the project provide many advantages for testing and simulation. Expanded information in the SCL file developed through this project has improved the ability to automate testing, and improve coverage. A few additional recommendations to encourage more thorough use of the standard have been identified through this work:

- The work of IEC 61850-6-2 is expected to improve the ability to graphically represent the single line diagram in a consistent manner. Having a consistent representation across tools from multiple vendors is not what is traditionally considered “interoperability”, but it will certainly help users interacting with different tools.
- SCL does not have an explicit description of a busbar. The convention of assuming a busbar from the method of interconnection is not reliable. Adding an SCL element for busbar and a particular type of connectivityNode would provide clarity²².
- Enforcing connection between SourceRef and ExtRef would improve the state of the system and allow understanding and therefore simulation of the connections in devices.
- Documenting the PowerSystemResource would improve the state of the system and allow understanding and therefore simulation of the connections in devices.²³

²² Recommendation for WG10

²³ Recommendation for users

- Use of SourceRef to document internal connections as well as GOOSE signals will improve the ability to simulate the device functionality.²⁴
- Using standardized IEC61131 logics to describe system behaviour at specification and implementation level improves the ability to automate testing and provide more rigorous simulation. At implementation level the IEC61850-90-11 provides ways to do this, it is recommended for WG10 to also enable this at specification level.²⁵

And a few improvements to the standard are recommended:

- The ability to characterize the input (active high/low, triggering, etc.)
- Adding standard semantics to the inputs of specific LNs so that the behaviour can be better documented.

And some product selection improves testing and simulation:

- Devices designed with the concepts of IEC 61850 structured functionality rather than having a 61850 communications interface “bolted on” tend to provide better testing and simulation. An example of this type of “bolted on” interface is shown in IED P1 SAMU on figure Figure 18 – Example of extRefAddr allowing documentation of deviation in receiving LN and function LN versus a direct interface shown in Figure 17
- Example of SCL specification and implementation of extRefAddr
- The simulation of the SSD provides a reference communication pattern that can be used for comparison of the real device behaviours.

9 Modelling

9.1 Battery data modelling

WP7.1 Aims at demonstrating a use case of energy storage regulation related to power transmission (originally proposed by RTE), in order to

- Identify potential gaps in the current IEC61850-7-420 datamodel related to energy storage
- Demonstrate the possibilities of offline testing and simulation of a storage regulation use case

This deliverable contains the use case description and demonstration outcomes. Deliverable 7.2 contains the demonstrator test and simulation results.

Given the only recent publication of IEC61850-7-420 in 2020, this datamodel is not yet widely known and implemented on the market. Many available DER technology manufacturers use

²⁴ Recommendation for users

²⁵ Recommendation for WG10

communication protocols and datamodels. In order to help pushing the use of IEC61850 in the DER industry, it is recommended to WG10 to provide simple profiles, which can easily be implemented by parties and manufacturers new to the IEC61850 standard. In this way bridges can be built between substation protection and automation systems and other systems interacting with it.

For the use case, the following setup is applied: 2 interconnected substations, with each a battery storage system connected to the busbar. The battery storage system is used to optimize the energy flow off the interconnection, in order to avoid overloading.

In the image below, we consider energy flowing from substation A to B.

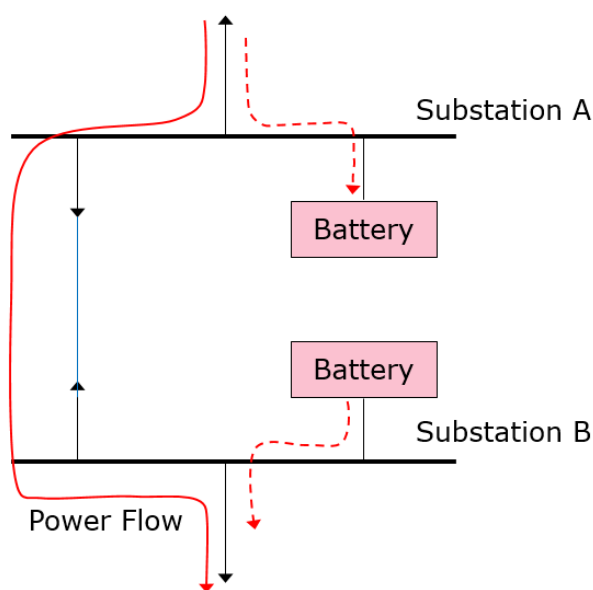


Figure 27 - Energy flow between substation A and B

The storage application shall be used to avoid overload situations on the interconnection. If the load is above a threshold, the surcharge will be stored in the battery of the substation that is the source of the energy flow and the battery in the other substation will supply that amount of energy into the network.

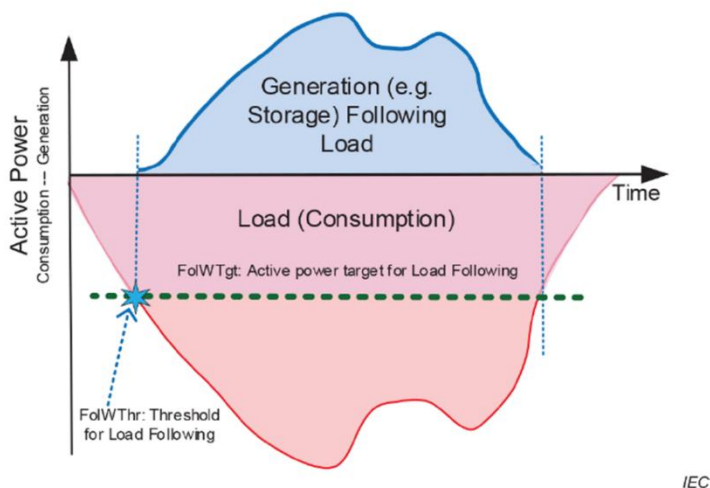


Figure 28 - Energy flow regulation principle (from IEC61850-7-420)

The initial state of charge of the batteries will be 50%. During an overload situation, the energy in one battery will increase and the one in the other battery will decrease. Once the load on the line is back beyond the threshold, the battery on the source side of the energy flow will feed energy up to the gap to the threshold into the network and the battery on the other side will consume that energy until the state of charge is again 50%.

The storage management function at the sink takes the lead and determines the current flow out of the storage. It will communicate with the storage system on the source side to determine the current flow into the storage system on the other side.

The algorithm for storage management requires the sum of all currents measured from the other feeders as an input. For the purpose of the demonstrator, that value will be simulated.

For creation of the use case datamodel, 3 main blocks can be identified:

- **Operational functions** translating the incoming data into requests for the power management system
- A **power management function** analyzing and prioritizing the requests from the operational functions and evaluate them against the DER capabilities and status.
- A **DER resource**, to provide the interface between the storage system and the power management function

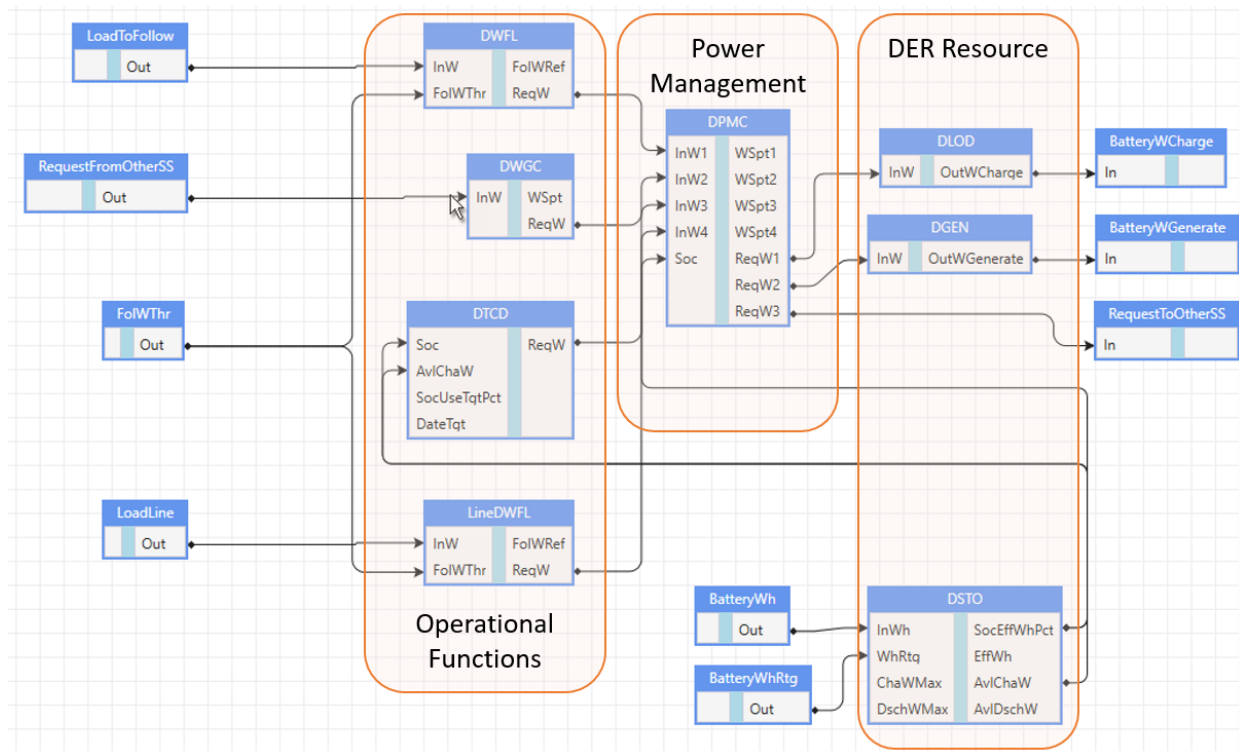


Figure 29 - Storage management system dataflow

9.1.1 Operational functions

The operational functions are used as follows:

9.1.1.1 DWFL

This is the load following function as illustrated in Figure 29 above. The function observes the power flow at a virtual reference point, which represents the sum of the power flow on all the other feeders.

- The DO FoIWRef provides the measured source / load at the reference point – so this is the load that is the sum of all the feeders. A negative value is a load
- The DO ReqW is the requested active power to be generated
- The DO FoIWThr is the setpoint that shall be set as a negative value of the rated power of the line – if the value of FoIWRef goes beyond this value, the difference will be requested in the DO ReqW

9.1.1.2 DWGC

This implements the requested power from the other substation. It applies when the other substation is driving.

9.1.1.3 DTCD

This is used to bring the battery back to the initial state of charge of 50% when there is no overload situation on the line anymore.

- The DO ReqW is the requested active power to be generated (positive value) or consumed (negative value).

- The DO SocUseTgtPct will be set to 50%
- The DO DateTgt will be set to "0" which shall be interpreted as "as soon as possible". It is recommended to WG10 to add a DO SocDurTmms (ING) as an alternate setting to DateTgt.

The function will use the indication from LN DSTO (AvlDschW / AvlChaW) to determine the power to charge or discharge.

9.1.1.4 LineDWFL

This load following function is used to prevent overloading of the line when the DTCD is active, bringing the battery back to 50% SOC. The function observes the power flow over the line.

- The DO FoWRef provides the measured source / load at the reference point – so this is the power flowing over the line. A negative value is a load
- The DO ReqW is the requested active power to be generated – in that case it indicates the maximum value that is allowed to charge the battery.
- The DO FoWThr is the setpoint that shall be set as a negative value of the rated power of the line – if the value of FoWRef goes beyond this value, the difference will be requested in the DO ReqW

9.1.2 Power management function – LN DPMC

The power management function handles the different requests from the operational functions and it provides the outputs towards

- DGEN: the requested active power to be generated (positive value)
- DLOD: the requested active power to be consumed to charge the battery (positive value)
- The battery on the other side of the line: the requested active power to be consumed (negative value) while the line is being protected from overload or to be generated (positive value) while the batteries are moving back to target SOC

The power management function is handling the input request as follows:

- If DWFL is > 0, it requests DGEN to generate the requested power as long as SOC is above 5% and the requested power is less than the max power discharging rating of the battery and it sends a request to the other battery to consume the same amount of power.
- If DWFL is 0, and DWGC requests a negative power, it requests DLOD to consume that power as long as the SOC is below 95%.
- If DWFL is 0, DWGC 0 or larger, and DTCD requests charging, it requests DLOD to consume the value requested by DTCD as long as the request from LineDFWL is 0. If the request from LineDFWL is not 0, the value requested by LineDFWL will be applied. In any case, it sends a request to the other battery, to produce the same amount of power

Additionally, the request to charge / discharge is using a ramping function to avoid immediate changes.

9.1.3 The resource LNs

9.1.3.1 Generic storage – DSTO

This contains the ratings for the storage device

- WhRtg – the rated battery capacity
- ChaWMax and DschWMax – the max power to be applied for charging or discharging the battery

Outputs produced are

- The state of charge in percentage of the effective available storage capacity (SocEffWhPct)
- The effective available storage capacity (EffWh)
- The effective available power to charge or discharge (AvlDschW / AvlChaW)

9.1.3.2 Generic Generator – DGEN

This LN receives the request from the power management function to generate power and discharge the battery.

9.1.3.3 Generic Load – DLOD

This LN receives the request from the power management function to consume power and charge the battery

9.2 Demonstrator data modelling gaps

9.2.1 Autorecloser use case

This chapter describes a IEC61850 modelling use case, explaining the modelling of a basic autorecloser function. Different modelling solutions are proposed with their pros and cons, some containing suggestions for standard enhancements (datamodel namespace extension).

It is requested that this use case is discussed in WG10 and leads to guidelines and/or standard extensions/modifications in order to standardise the approach of this type of use cases.

For this use case a HV interconnection is imagined protected by a distance protection (with different protection zones) and equipped with an autorecloser and synchrocheck (only for 3ph autoreclosing) function.

9.2.1.1 Use case Setup

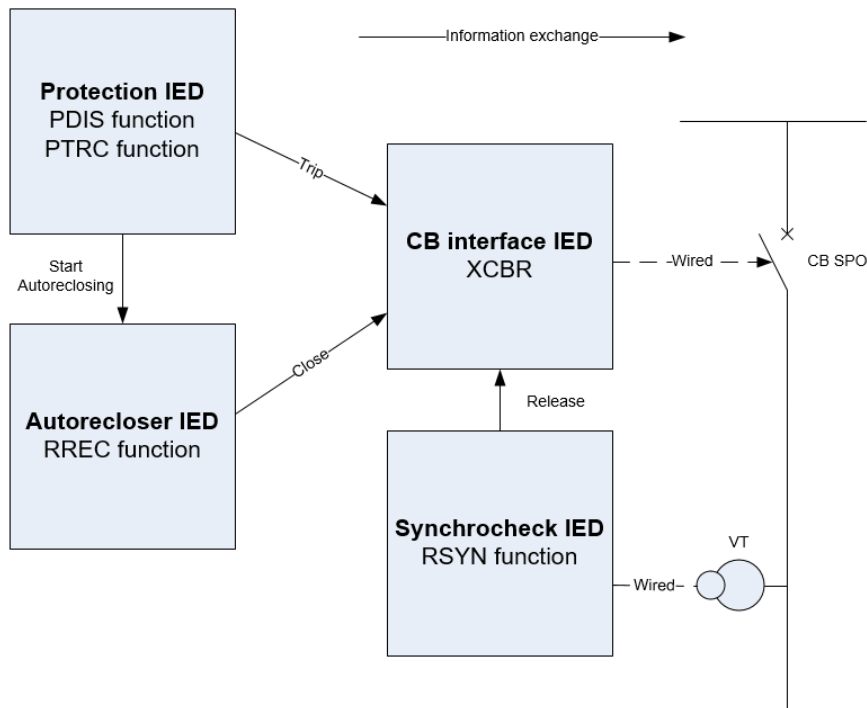


Figure 30 - Use case description

Protection IED:

IED containing a distance protection function with different zone settings.

PDIS zone 1 = PDIS instance covering the internal faults with protection time setting $t = 0s$

The IED also contains a trip logic represented by a PTRC function, collecting all trips and sending a trip signal to the circuit breaker interface.

Autorecloser IED:

IED containing the autorecloser RREC function

The autorecloser function is equipped with different dead time settings for single and multiphase trips, and can only be activated for internal faults of the interconnection.

Only 1 autoreclosing attempt is applied, where after a reclaim time is applied. After the reclaim time, the autorecloser function will accept new activations.

Autoreclosing can only be activated, when an internal fault has been detected, and the trip signal to the circuit breaker has been issued.

The autorecloser function is not able to define, based on the protection and trip signals, if the fault is single or multiphase. The autorecloser function needs a specific activation of the single phase or multiphase sequence by the protection IED.

In case of multiphase autoreclosing, synchronisation needs to be checked in order to close the CB

CB interface IED:

Providing the IEC61850 interface of the circuit breaker

Synchrocheck IED:

IED containing synchrocheck function, providing a release to the autorecloser function, for the closing of the circuit breaker.

9.2.1.2 Modelling Options

9.2.1.2.1 Modelling Variant 1 – As demonstrated in Osmose demonstrator

Create specific PDIS and PTRC instances for single or multiphase faults in order to differentiate at the source if a trip is single or multiphase. Use a specific GAPC to activate single or three phase autoreclosing.

Pros:

- The current IEC61850-7-4 datamodel can be used. No namespace extensions are required

Cons:

- Protection IEDs need to have the possibility to be configured with specific PDIS instances for single and multiphase faults
- More complex configuration of the distance protection function in the distance protection IED

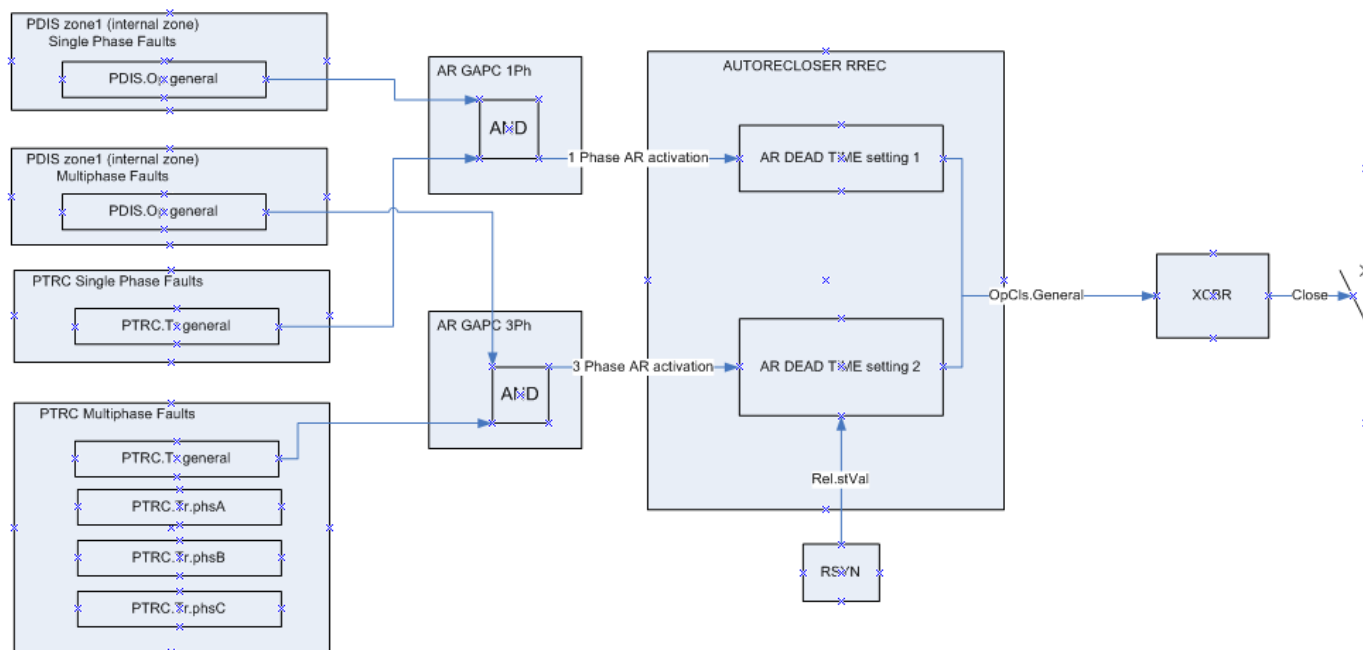


Figure 31 - Variant 1

9.2.1.2.2 Modeling Variant 2

Extend PDIS and PTRC LNs with specific data objects (SPS) for single and multiphase trips. Use a specific GAPC to activate single or three phase autoreclosing.

Regarding the datamodel extension there are 2 possible solutions:

1. Extend with data-objects for each possible combination

StrAN / StrBN / StrCN / StrAB / StrBC / StrAC

OpAN / OpBN / OpCN / OpAB / OpBC / OpAC

TrAN / TrBN / TrCN / TrAB / TrBC / TrAC

2. Extend with data objects which define if it's a single or multipole event.

Str1Ph / Str3Ph

Op1Ph / Op3Ph

Tr1Ph / Tr3Ph

Pros:

- No need to multiply PDIS and PTRC instances, straight forward distance protection configuration in the protection IED
- Datamodel extensions cover also the needs of other use cases (see 4)

Cons:

- Extension of the protection and trip LN data model is needed

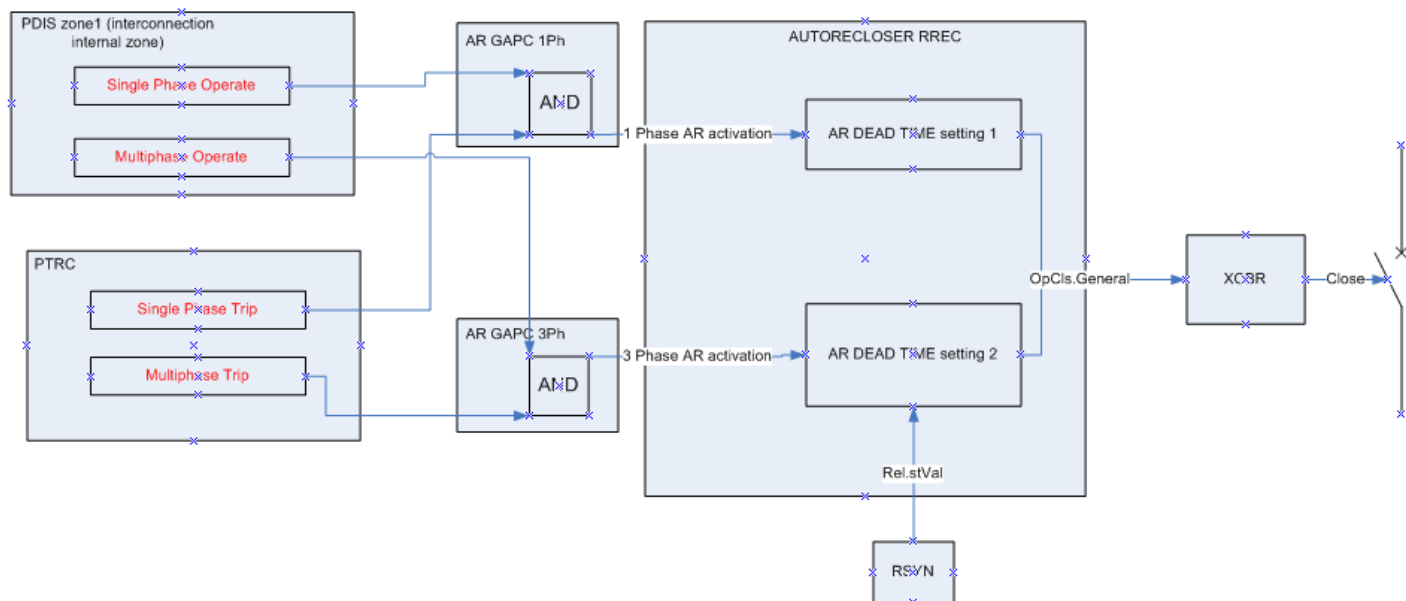


Figure 32 - Variant 2

A possible variant of this solution is to extend the RREC LN with the content of the proposed GAPCs.

9.2.1.2.3 Modelling Variant 3

Create a dedicated GAPC LN with a logic which takes the signals coming from the protection and trip LNs and creates the needed signals for the autorecloser function

Pros:

- The current IEC61850-7-4 datamodel can be used. No namespace extensions are required (standard GAPC model is used)
- A single generic logic could be created and instantiated multiple times

Cons:

- More modelling and configuration is needed (creation of an extra custom logic)

The image below describes the principle and a detail of the single phase detection function GAPC. The multiphase detection function GAPC has a similar behaviour.

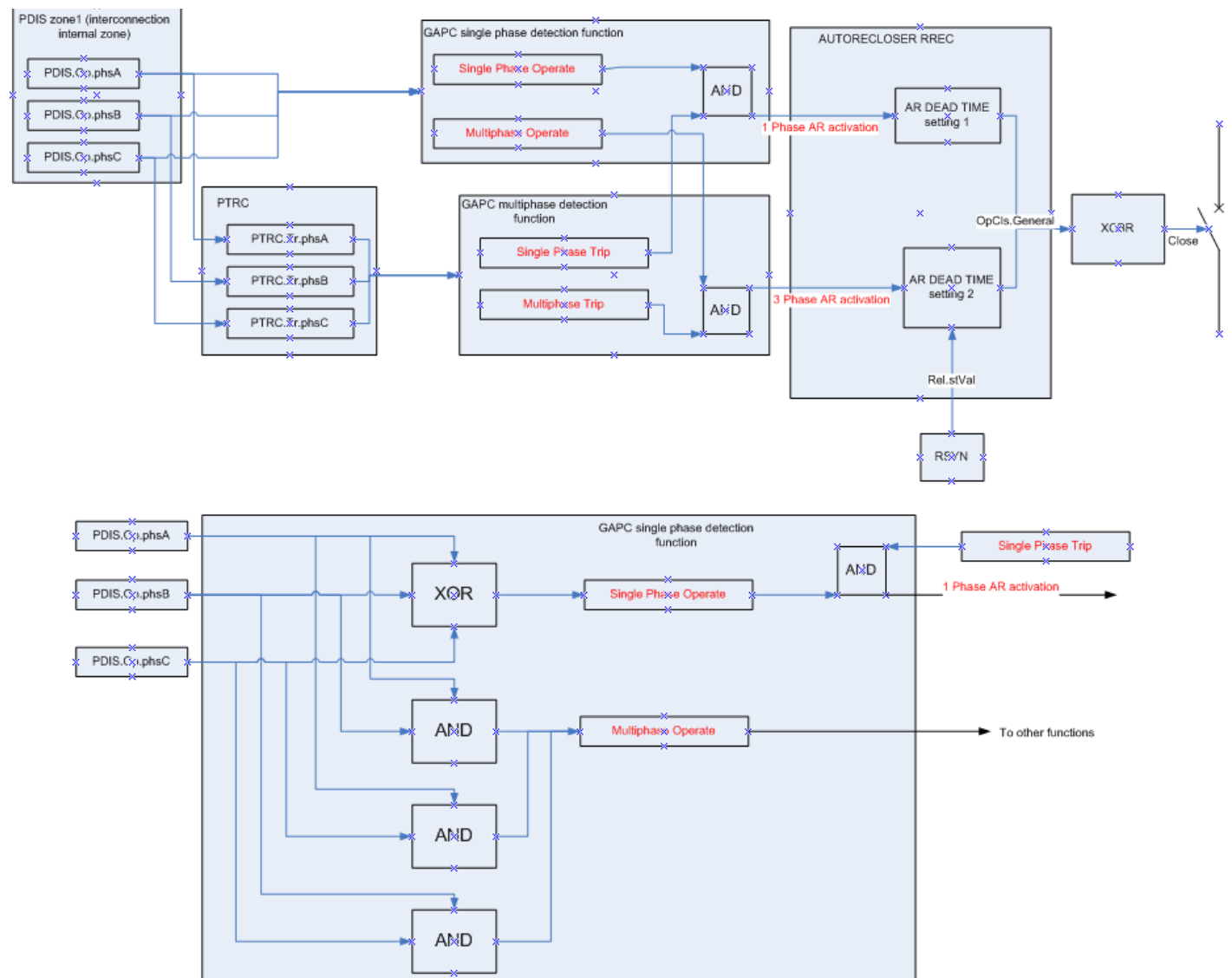


Figure 33 - Variant 3

9.2.1.2.4 Modelling Variant 4

Extend PTRC with the requested model for autorecloser interface

Pros:

- No GAPC needs to be created
- Only 1 LN needs to be extended with new logic/datamodel

Cons:

- Extra logics need to be integrated in PTRC
- Datamodel extension covers only trips, not the operate outputs from the protection functions. This limited solution does possibly not cover all use cases.

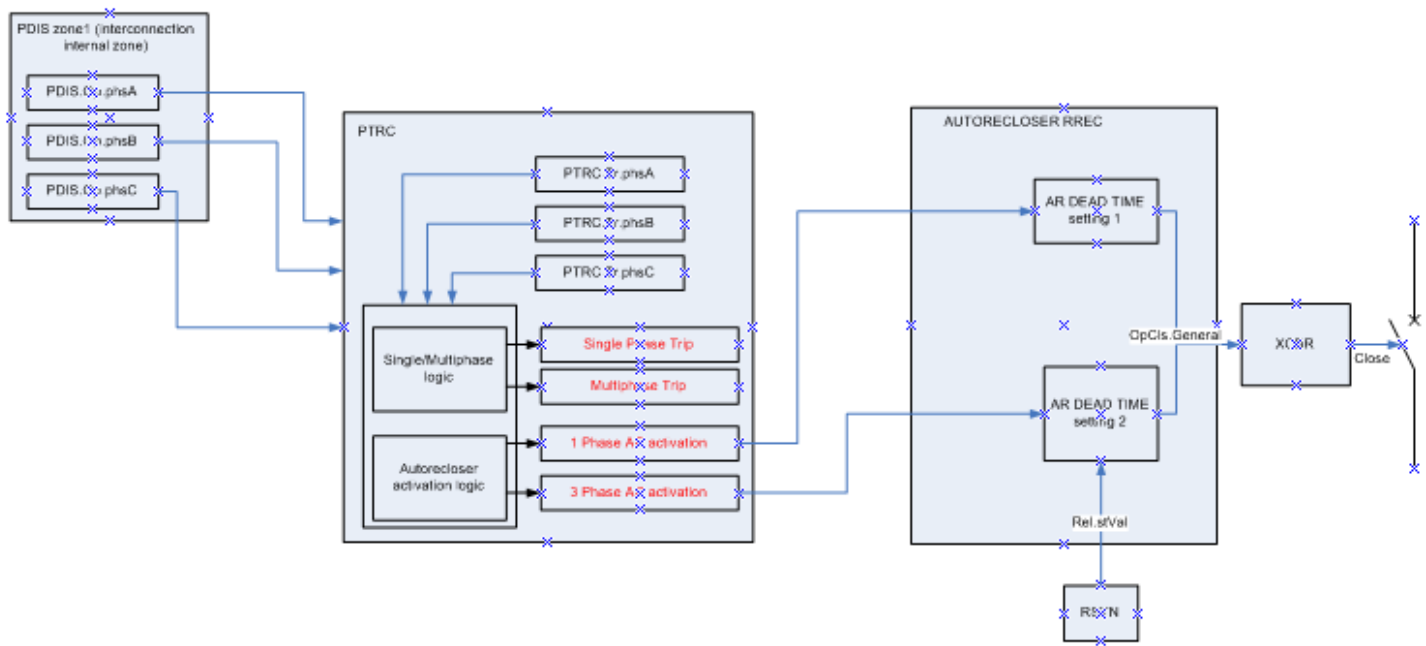


Figure 34 - Variant 4

9.2.1.2.5 Other possible use cases

There are some other protection and automation functions which behave differently for single phase and multiphase faults. For these functions, similar use cases can be described.

1Ph/3Ph Zone extention: extension of the internal protection zone of an interconnection when teleprotection and autorecloser functions are in service and working correct. This function needs to identify if a fault is single or multiphase in order to apply the correct behavior.

Trip logic: The trip logic monitors the autorecloser function in order to decide if a single phase fault results in a single phase or 3phase trip. When the autorecloser is out of service, all single phase faults need to lead to a three phase trip.

Backup function: the protection back-up function (which trips all busbar connected circuit breakers if a single bay circuit breaker fails to trip) behaves differently depending on the type of fault.

9.3 Additional modelling recommendations

9.3.1 Modelling of physical resources

In order to allow a more granular specification of dataflow and detailed specification of an IED, it is recommended for WG10 to extend the IEC61850-7-4 datamodel with additional LNs representing the physical interfaces of an IED. These LNs may represent elements of the IED:

- Binary inputs
- Binary outputs
- Analogue inputs
- Analogue outputs
- LEDs
- Function keys
- Power supply

This also provides additional opportunities related to automated generation of documentation of the IED configurations, related to the IEDs physical interfaces.

9.3.2 Teleprotection GOOSE tunnelling

It is recommended for WG10 to clarify how to model in the SCD file the communication section for the usecase where 2 devices from different substations exchange GOOSE signals by applying tunnelling. Within Osmose the following solution was applied:

- Adding the remote IED to the local subnetwork, without IP parameters but with definition of the GOOSEs
- Using the desc attribute of the ConnectedAP to indicate the role of this specific remote IED in the SCD file

10 Annexes

10.1 A: Interoperability test specification (ID7.2.A)

This annex is attached to deliverable D7.2 Execution of demonstrator for interoperability framework

10.2 B: SCL comparison specification (ID7.3.B)

10.3 C: Enhancements of SCL for virtual signal flow and virtual IED (ID7.3.C)

10.4 D: Annex D - IEC_61850-6-100.XSD.2019A6.WD.2021-08-24 (ID7.3.D)

11 Overview of all recommendations

Chapter	Recommendation topic	For
3	Acknowledgment of the role of a system specification tool	WG10
3.1	SCL Extensions as defined in Annex D	WG10
3.2.1	LNode datamodel namespace	SST
3.2.1	.fsd concept and format	WG10
3.2.1	function definition SCL file location	WG10
3.2.1	subfunction attribute definition	WG10
3.3	Specification process support	SST
3.3	Functional physical allocation of (sub)functions	WG10
3.3	Virtual IED concept	WG10
3.3	(sub)function (group) behavior description concept	WG10
3.3.1.1	GUID element concept	WG10
3.3.1.2	DataTypeTemplate corrections	SST
3.3.1.3	LNode datamodel namespace	WG10
3.3.1.3	Datamodel namespace upgrade	SST
3.3.3	Copying of private elements in (sub)function instances	SST
3.3.3	Creating LNode uniqueness	SST
3.3.3.2	PowerSystemResource element	WG10
3.3.4	Implementation of dataflow from specification	WG10
3.3.4	Specification of internal dataflow	Users
3.3.4	Modification of function instance inputs	SST
3.3.4	Use of the "input" attribute	WG10
3.3.4	Rules regarding extRefAddr semantic	WG10

3.3.4	Introduction of a "servicespec" attribute	WG10
3.3.5	Predefined rules for virtual IED creation	WG10
3.3.5	Enhancement of the IED service section	WG10
3.3.5	Support of variant scenarios	WG10
3.3.6	SCL definition of a template or instance	WG10
3.4.1	IED selection scenario support	SCT
3.4.2	Creation of an ICD with enhanced substation section	ICT
3.4.3	IED selection scenario support	SCT
3.4.3.1.4	ISD service section content	WG10
3.5	ExtRef IED section location	IED / ICT
3.5	Specification of logical behavior	WG10
3.7	version and rights management	WG10
4.1	ISD/ICD content	WG10
5	Use of comparison tools	Market
6	SCL content for SSD exchange between substations	WG10
7	Documentation	SST/SCT
8	Busbar process element	WG10
8	PowerSystemResource documentation	Users
8	Definition of internal dataflow	Users
8	Specification of logical behavior	WG10
9.1	IEC61850-7-420 datamodel implementation	WG10
9.1.1.3	IEC61850-7-420 datamodel extension	WG10
9.2.1	Autorecloser usecase datamodelling GAP	WG10
9.3.1	Physical resources	WG10
9.3.2	Teleprotection GOOSE tunneling IED SCL location	WG10

