
Implementation Guidelines – CGMES v2.4.15

15 March 2016

This document provides implementation guidelines for Suppliers and TSOs on the ENTSO-E Common Grid Model Exchange Standard (CGMES) version 2.4.15.

The clarifications provided in this document are considered as required rules for the implementation of the CGMES v2.4.15. Issues related to the Dynamics profile are not discussed in this document.

The content of the implementation guide will be reflected in the following CGMES version.

The document was extensively discussed with vendors and TSOs on 22 June 2015 and put for consultation until 29 June 2015. The issues on LTCflag and the following were added in Feb-Mar 2016.

Table of contents

Table of contents.....	2
1 Introduction	4
2 TapChanger.neutralU vs PowerTransformerEnd.ratedU vs VoltageLevel.BaseVoltage	4
2.1 Issue description	4
2.2 Required implementation	4
3 ACLineSegment-s between different terminal voltages	5
3.1 Issue description	5
3.2 Required implementation	5
4 Association from ConformLoadGroup/NonConformLoadGroup	6
4.1 Issue description	6
4.2 Required implementation	6
5 LTCflag	7
5.1 Issue description	7
5.2 Use cases	8
5.2.1 Power flow calculation relies on SSH	8
5.2.2 Power flow calculation relies on SV	8
5.2.3 Power flow calculation relies on either SSH or SV	8
5.3 Required implementation	8
6 GeographicalRegion, SubGeographicalRegion	10
6.1 Issue description	10
6.2 Required implementation	10
7 GeneratingUnit.normalPF	10
7.1 Issue description	10
7.2 Required implementation	10
8 Transformer parameters	10
8.1 Issue description	11
8.2 Required implementation	11
9 The sign of the PhaseTapChangerTablePoint.angle	11
9.1 Issue description	11
9.2 Required implementation	11
10 Slack generator	12
10.1 Issue description	12
10.2 Required implementation	12
11 SynchronousMachine.qPercent	12
11.1 Issue description	12

11.2 Required implementation 12

1 Introduction

This document provides guidelines for Suppliers and TSOs that are implementing CGMES version 2.4.15. The need to produce such document is justified by the fact that during the implementation of the CGMES since Dec 2013 there were some issues reported but were not fully clarified in the last released version of the CGMES, i.e. version 2.4.15.

The CGMES conformity process and the tests that are performed by TSOs using real network models reveal that there are different implementations by the Suppliers. The present document defines required implementation for some of the CGMES classes/attributes in order to achieve interoperability.

2 TapChanger.neutralU vs PowerTransformerEnd.ratedU vs VoltageLevel.BaseVoltage

2.1 Issue description

The following questions were addressed to clarify the use of TapChanger.neutralU and PowerTransformerEnd.ratedU:

- 1) Does RatioTapChanger.stepVoltageIncrement relate to TapChanger.neutralU or PowerTransformerEnd.ratedU?
- 2) When and why can PowerTransformerEnd.ratedU differ from TapChanger.neutralU?
- 3) Is TapChanger.neutralU relevant for PhaseTapChanger (PhaseTapChangerAsymmetrical, PhaseTapChangerNonLinear, PhaseTapChangerSymmetrical and PhaseTapChangerLinear)?
- 4) Is SvTapStep.position mandatory in the SV file when a transformer has TapChanger.controlEnabled = false?

These questions were discussed within ENTSO-E and IEC-TC57-WG13. The discussion on the different items, which are mentioned above, can be summarized as follows:

1)

It should be taken into account that the description is: "Tap step increment, in per cent of nominal voltage, per step position." Please also note that for PhaseTapChanger it is explicitly mentioned that voltageStepIncrement relates to "nominal voltage of the transformer end" = PowerTransformerEnd.ratedU. Hence, the neutralU discussion is then only relevant for RatioTapChanger.

The advice from WG13 is that on RatioTapChanger.stepVoltageIncrement should be neutral voltage, not nominal. WG13 will write a CIM issue to clarify this in the upcoming standards. On PhaseTapChangerNonLinear.voltageStepIncrement description is currently "nominal voltage of the transformer end". WG13 proposes to change this to "neutral voltage of the tap changer".

2)

WG13 advises that TapChanger.neutralU is the voltage that will be at the neutral step. Conventions can differ.

3)

WG13 advises that it is only relevant for PhaseTapChangerNonLinear.

4)

WG13 advises that SvTapStep.position should be mandatory. WG13 will prepare an update of the IEC61970-456.

2.2 Required implementation

Considering the outcome of the ENTSO-E discussion, consultations with WG13 and concerned suppliers implementing CGMES, ENTSO-E requires the following implementation:

- TapChanger.neutralU is the voltage at the terminal of the PowerTransformerEnd associated with the tap changer when all tap changers on the transformer are at their neutralStep position. Normally neutralU of the tap changer is the same as ratedU of the PowerTransformerEnd, but it can differ in special cases such as when the tapping mechanism is separate from the winding more common on lower voltage transformers. For CGMES neutralU equals ratedU.
- RatioTapChanger.stepVoltageIncrement shall be in per cent of neutral voltage, per step position, not nominal. The right description of this attribute is: Tap step increment, in per cent of neutral voltage, per step position.
- Nominal quantities are not related to the equipment but to the system nominal voltage in the grid.
- Rated quantities such as ratedU are related to the nameplate data.
- TapChanger.neutralStep is the step position where the voltage is neutralU when the other terminals of the transformer are at the ratedU. If there are other tap changers on the transformer those taps are kept constant at their neutralStep.
- For PhaseTapChangerAsymmetrical, PhaseTapChangerSymmetrical and PhaseTapChangerLinear the neutralU is not relevant.
- PhaseTapChangerNonLinear.voltageStepIncrement relates to the ratedU. The description will be updated by IEC/WG13. The voltageStepIncrement is used for PhaseTapChangerAsymmetrical, PhaseTapChangerSymmetrical.
- SvTapStep.position shall be required even if the TapChanger.controlEnabled is set to false.

As CGMES requires neutralU to be equal to ratedU, the following implementations are valid:

- $\text{ratedU} + \text{ratedU} * \text{stepVoltageIncrement} * (\text{SvTapStep.position} - \text{neutralStep})$
- $\text{neutralU} + \text{neutralU} * \text{stepVoltageIncrement} * (\text{SvTapStep.position} - \text{neutralStep})$
- $\text{neutralU} + \text{ratedU} * \text{stepVoltageIncrement} * (\text{SvTapStep.position} - \text{neutralStep})$
- $\text{ratedU} + \text{neutralU} * \text{stepVoltageIncrement} * (\text{SvTapStep.position} - \text{neutralStep})$

3 ACLineSegment-s between different terminal voltages

3.1 Issue description

Different implementations by the Suppliers for lines connecting nodes having different nominal voltage lead to different flows on lines that have same physical characteristics. This happens due to the fact that some implementations are using different base voltage when converting the physical units of the line to per unit system and vice versa. This is especially valid when multiple parties are importing and re-exporting between different tools.

In order to resolve this issue CGMES version 2.4.15 clearly specifies that “Each ACLineSegment is required to have an association to a BaseVoltage. The association to Line is not required. ENTSO-E exchanges allow 10 % difference of the BaseVoltage.nominalVoltage at the two ends of an ACLineSegment representing a complete tie-line or connecting to a boundary node”

OCL constraint on this was also added to CGMES.

3.2 Required implementation

Considering the outcome of the ENTSO-E discussion, consultations with WG13 and concerned suppliers implementing CGMES, ENTSO-E requires the following implementation:

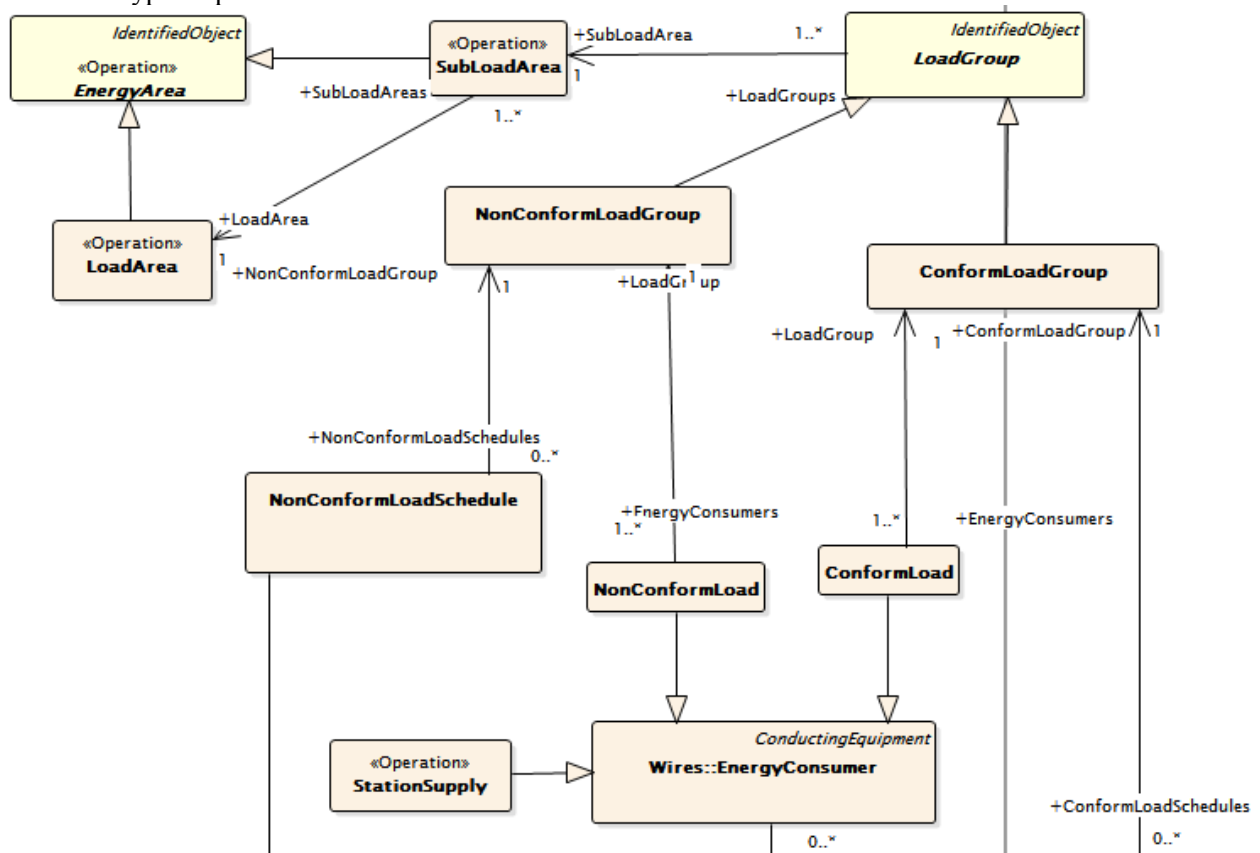
- All implementations shall use association to a BaseVoltage for the purpose of any per unit calculations and shall not rely on the voltages (neither nominal nor actual values obtained by previous or current solution) at the nodes, which the ACLineSegment connects to.

- In case there are interconnected ACLineSegments with different BaseVoltage for different parts of the networks (when assembling different model authority sets) the application needs to handle this to ensure accurate physical units.

4 Association from ConformLoadGroup/NonConformLoadGroup

4.1 Issue description

SubLoadArea is marked as “Operation”. Classes ConformLoadGroup and NonConformLoadGroup inherit the association to SubLoadArea from LoadGroup. Therefore, it is logical that the association is also marked with stereotype “Operation”.



4.2 Required implementation

Considering the outcome of the ENTSO-E discussion, consultations with WG13 and concerned suppliers implementing CGMES, ENTSO-E requires the following implementation:

- The association from ConformLoadGroup and NonConformLoadGroup to SubLoadArea is required for “Operation”, but optional for Base.
- The association “SubLoadArea” for ConformLoadGroup and NonConformLoadGroup is considered as “Operation” and will be marked in the next version of the CGMES.

ENTSO-E would like to inform that CIMdesk application have implemented the following validation rules:

- Each ConformLoad and NonConformLoad shall be associated with ConformLoadGroup and NonConformLoadGroup. This is for both “Equipment Core” (Base) and “Operation’ validation.

- Association from ConformLoadGroup and NonConformLoadGroup to SubLoadArea is required for “Operation”, but optional for Base.

5 LTCflag

5.1 Issue description

The problem is related to the attribute TapChanger.ltcFlag (description of the attribute is “Specifies whether or not a TapChanger has load tap changing capabilities.”) and the association TapChanger.TapChangerControl.

Previously agreed rules are the following:

- If TapChanger.ltcFlag is true, and TapChanger.TapChangerControl is not provided, Error: Missing association TapChanger.TapChangerControl, required if attribute TapChanger.ltcFlag is 'true'.
- If TapChanger.ltcFlag is false, and TapChanger.TapChangerControl is provided, Warning: "Association TapChanger.TapChangerControl is ignored, because attribute TapChanger.ltcFlag is 'false'.
- If TapChanger.ltcFlag is false, and TapChanger.TapChangerControl is not provided, Alert: "The TapChanger is fixed."
- If TapChanger.ltcFlag is true, TapChanger.TapChangerControl is provided, and TapChanger.controlEnabled is false, Alert: "The TapChanger control is disabled."
- If TapChanger.ltcFlag is false, and TapChanger.controlEnabled is true, Warning: "The control is enabled for a fixed TapChanger."
- If TapChanger.controlEnabled is true, and SvTapStep is not provided, Error: "SvTapStep is not provided, required if TapChanger.controlEnabled is true."

The above rules were based on the interpretation that:

- TapChanger.ltcFlag is used to indicate whether there is physical tap changer controller installed. For a fixed or a manually controlled transformer, this flag is ‘false’. Then the association to TapChanger.TapChangerControl is not needed.
- TapChanger.controlEnabled is a flag indicating that whether the installed tap changer is actually in-service. You could have an installed tap changer, but not put it into the service. In other words, the tap changer is disabled. In that case, the association to TapChanger.TapChangerControl is still needed. The control scheme is just not enabled.

It was assumed that TapChanger.ltcFlag is not only used for specifying tap change controlling capability. It is not sure whether (voltage/active power regulating) load tap changing (LTC) is different from regular (manual) tap changing. That might be the case as there are no flags similar to TapChanger.ltcFlag in other voltage regulating equipment (ShuntCompensator, etc.). If that is the case, there is a need to decouple attribute TapChanger.ltcFlag from association TapChanger.TapChangerControl.

Based on the above rules it is no longer possible to export a transformer with TapChanger.ltcFlag=true and without given TapChanger.TapChangerControl. There are significant number of transformers which are modelled with on-load tap changers which are switched manually. These are currently modelled with on-load tap changing capability, but without controller. Nevertheless, in the past these transformers have been modelled with a minimum voltage and a maximum voltage at the controlled bus section, but with the control enabled flag set to false (TapChanger.controlEnabled=false) in the legacy systems.

The TapChanger.ltcFlag option is also relevant and used for the IEC short-circuit calculation, so vendors cannot reset the option to "false", because this leads to incorrect short-circuit results. Further, a creation of a TapChanger.TapChangerControl is also not preferred, because a tap-controller does not exist in reality. It is possible however, but not desirable because the control data would be fake data.

5.2 Use cases

The following use cases are defined assuming that different datasets of the instance data that is exchanged with CGMES serve different purpose from operational planning to long term planning cases.

It is also assumed that in case the capabilities of the physical equipment are changing after a point in time the change will be properly reflected in the EQ and SSH profiles or the software enables the user to modify or make different use of this data during specific analyses.

Depending on the settings of the power flow algorithm different software can produce a different power flow results, e.g. tap positions may be different, the reactive injections at synchronous machines would be different, the reference generator (slack) active power will differ.

The use cases are defined for the purpose of tap regulation and may not cover all other aspects for operational or long term planning.

5.2.1 Power flow calculation relies on SSH

The calculation of a power flow taking EQ and SSH as an input is the most frequent use case for initial calculation. It is mostly used when a party (e.g. a TSO) is requested to provide unsolved model (EQ and SSH, for node-breaker model or EQ, SSH and TP for bus-branch model), or a solved model is to be prepared (EQ, SSH, TP and SV).

In this situation, the software makes use of classes and attributes in EQ and SSH to perform the topology process and run power flow calculation. The outcome is TP (from the topology processing function) and SV (from the power flow calculation). Tap regulation could be essential for converging of the power flow algorithm and for achieving good voltage profile. Therefore, the power flow algorithm normally provides capabilities to the user to enable or disable tap regulation during power flow. This is normally a global setting which makes use of EQ and SSH information to apply or not apply tap regulation for different transformers. There is no modification of EQ and SSH data but just use of it. In case when the user needs to perform specific studies (e.g. modification of equipment to enable physical capability to regulate on load or to enable the regulation for a particular tap changer) he/she needs to perform set of actions on the EQ and SSH data so that the software is capable to consider the new situation.

5.2.2 Power flow calculation relies on SV

In some cases, it is required that the starting point of the power flow algorithm is a previously computed solution. This mainly applies for business processes in which an entity is using the data from multiple parties and providing service to all parties by performing assembling and power flow calculation for the assembled model. The outcome of the service is a single SV instance file representing the solution (e.g. for the ENTSO-E Common Grid Model).

In order to make use of this service individual parties that provided data would rely on the SV to reproduce exactly the same power flow solution. In case the power flow runs from SV the software will make use of tap positions (and other data) that are present in the SV data (as well as the data in EQ and TP).

5.2.3 Power flow calculation relies on either SSH or SV

This a combined situation in which the receiving software has access to both SSH and SV data and enables the user to select what kind of data will be the basis for the power flow calculation. In all cases no data is modified but it is just that the software uses one or another set of data.

5.3 Required implementation

Control of tap changers in a power flow type of application is made using the TapChangerControl class and the TapChanger.ltcFlag. If a TapChanger has a TapChangerControl (referenced TapChanger.TapChangerControl) means that the power flow application may control the tap changer. The TapChanger.ltcFlag provides information that the TapChanger has physical capability to move the tap under load. Also used in the IEC 60909 calculations to indicate if the tap can move on load.

The meaning of the combinations for TapChanger.TapChangerControl and TapChanger.ltcFlag are described in the following table **Error! Reference source not found.**

TapChanger.ltcFlag	TapChanger.TapChangerControl	Description
False	Not present	A real and fixed tap that is not controlled and cannot be moved on load (manual tap change). Power flow cannot be set to change the tap for voltage control during the calculation.
True	Not present	A real and fixed tap that is not controlled but can be moved under load, e.g. manually. Power flow cannot be set to change the tap for voltage control during the calculation. Optimal power flow might have access to these taps and change in order to optimize. Also State Estimator may estimate the tap position to find a better solution for the system state.
True	Present	A real tap with a possibility to change taps automatically for voltage control/active power (load flow) enabled. Depending on the RegulatingControl.enabled and TapChanger.controlEnabled in SSH, the power flow shall or shall not participate in the regulation. In cases where the RegulatingControl is associated with more than one tap changer or other devices the attribute TapChanger.controlEnabled can be set to false in order to set which of the taps are not enabled.
False	Present	An artificial tap changer can be used to simulate control behavior in power flow.

Please note that SvTapStep is required (always present in the SV) for all TapChangers.

CIMdesk validation is corrected to check the following:

- If TapChanger.ltcFlag is true, and TapChanger.TapChangerControl is not provided, Alert: “A real and fixed tap that is not controlled but can be moved under load, e.g. manually. Power flow cannot be set to change the tap for voltage control during the calculation. Optimal power flow might have access to these taps and change in order to optimize. Also State Estimator may estimate the tap position to find a better solution for the system state.”
- If TapChanger.ltcFlag is false, and TapChanger.TapChangerControl is provided, Alert “An artificial tap changer that can be used to simulate control behavior in power flow.”
- If TapChanger.ltcFlag is false, and TapChanger.TapChangerControl is not provided, Alert: “A real and fixed tap that is not controlled and cannot be moved on load (manual tap change). Power flow cannot be set to change the tap for voltage control during the calculation.”
- If TapChanger.ltcFlag is true, TapChanger.TapChangerControl is provided. Alert: “A real tap with a possibility to change taps automatically for voltage control/active power (load flow) enabled. Depending on the RegulatingControl.enabled and TapChanger.controlEnabled in SSH, the power flow shall or shall not participate in the regulation. In cases where the RegulatingControl is associated with more than one tap changer or other devices the attribute TapChanger.controlEnabled can be set to false in order to set which of the taps are not enabled.”
- If SvTapStep is not provided for all TapChangers, Error: "SvTapStep is not provided, required for all TapChangers."

6 GeographicalRegion, SubGeographicalRegion

6.1 Issue description

During the CGMES by TSOs it was found out that different applications have different logic for assignment of the GeographicalRegion and SubGeographicalRegion. This results in model which have as many GeographicalRegion and SubGeographicalRegion as number of substations are present in the model. Also the other extreme of having only one set of GeographicalRegion and SubGeographicalRegion is observed.

6.2 Required implementation

The following **best practice** implementation was agreed during the workshop on 3-4 March 2016.

- One GeographicalRegion should be exchanged per MAS. In case TSOs have a need to have the same GeographicalRegion (i.e. multiple TSOs in a country) the class GeographicalRegion shall be present in all TSO models and shall have different rdf:ID, but can have same name/description.
- SubGeographicalRegion is normally a TSO or sub-area of a TSO.

There is no specific naming convention as previously agreed for CGMES 2.4.

7 GeneratingUnit.normalPF

7.1 Issue description

During the CGMES by TSOs it was found out that different applications still have different understanding of GeneratingUnit.normalPF. In CGMES 2.4 it is stated that this attribute is used in case of distributed slack. However there is no statement on the values that are allowed for this attribute.

7.2 Required implementation

The following was agreed during the workshop on 3-4 March 2016.

- It is allowed to have values for GeneratingUnit.normalPF which sum is different than 1. The application that imports such model shall have a load flow calculation logic setup (for different MAS that use very different normalPF values the values must be normalized) to use the information for the purpose of the distributed slack and preserve the original values of GeneratingUnit.normalPF in case of export.
- Example of GeneratingUnit.normalPF values (before normalization) for different MAS

	MAS 1	MAS 2	MAS 3
GeneratingUnit.normalPF for Gen 1	0	200	50
GeneratingUnit.normalPF for Gen 2	0.3	500	60
GeneratingUnit.normalPF for Gen 3	0.4	150	45

8 Transformer parameters

8.1 Issue description

During the CGMES by TSOs it was found out that different applications have different understanding of TransformerEnd.end number and its relation with the parameters of the PowerTransformerEnd. As a result the values for the parameters of two winding transformers are exported/imported wrongly which causes significant interoperability issues.

8.2 Required implementation

The following was agreed during the workshop on 3-4 March 2016.

- The applications shall follow strictly the description of the PowerTransformerEnd which states “for a two Terminal PowerTransformer the high voltage PowerTransformerEnd has non zero values on r, r0, x, and x0 while the low voltage PowerTransformerEnd has zero values for r, r0, x, and x0.”
- The high voltage side is given by the TransformerEnd.endNumber: “Number for this transformer end, corresponding to the end's order in the power transformer vector group or phase angle clock number. Highest voltage winding should be 1. Each end within a power transformer should have a unique subsequent end number. Note the transformer end number need not match the terminal sequence number.”
- In case of a two winding transformer with same rated voltage (PowerTransformerEnd.ratedU) on both sides the application makes sure that only one side has TransformerEnd.endNumber equals to 1.
- Therefore the parameters are always provided for PowerTransformerEnd which has TransformerEnd.endNumber equal to 1.

9 The sign of the PhaseTapChangerTablePoint.angle

9.1 Issue description

In the frame of the implementation of CGMES 2.4 a problem with the interpretation of the PhaseTapChangerTablePoint.angle was found. This causes different results in the load flow.

The following questions were addressed to IEC 57/WG13:

- Does the interpretation of the attribute PhaseTapChangerTablePoint.angle depend on the TransformerEnd.endNumber?
- What is the relation between voltage angles on no-load conditions in the particular example?

9.2 Required implementation

The following feedback was received from WG13 on 28 Oct 2015:

- The only place in the CIM UML where the meaning of a positive phase shift is specified is on the description of the attribute PhaseTapChangerLinear.stepPhaseShiftIncrement. That says “A positive value indicates a positive phase shift from the winding where the tap is located to the other winding (for a two-winding transformer).”
- It would be good to keep a consistent definition throughout the CIM UML.
- It is agreed to add this explanation of positive phase shift to the description of the class “PhaseTapChanger”.
- It would also be good to add an example to 61970-301.

Therefore, `PhaseTapChangerTablePoint.angle` in CGMES 2.4 shall be implemented considering that “A positive value indicates a positive phase shift from the winding where the tap is located to the other winding (for a two-winding transformer).”

10 Slack generator

10.1 Issue description

The issue on defining slack generator was addressed in the IOP 2014. The modifications applied in the CGMES v2.4.15 aimed to clarify how the slack generator is defined. However, it was found that still some vendors have different understanding. Therefore, the implementation guide points out some of the key references in the CGMES 2.4.15 documentation.

10.2 Required implementation

`SynchronousMachine.referencePriority` is used to define the slack generator - reference angle. `SynchronousMachine.referencePriority` is a required attribute in the SSH.

In SSH

`SynchronousMachine.referencePriority` - Priority of unit for use as powerflow voltage phase angle reference bus selection. 0 = don't care (default) 1 = highest priority. 2 is less than 1 and so on.

Please note that `GeneratingUnit.normalPF` is used for representing distributed slack participation factor.

In SV

`TopologicalIsland.AngleRefTopologicalNode` - The angle reference node is the `TopologicalNode` to which a synchronous machine is connected and referenced from `TopologicalIsland.AngleRefTopologicalNode`. Therefore, this is the `TopologicalNode` to which a `SynchronousMachine` is connected and has `SynchronousMachine.referencePriority` equals 1 in SSH.

11 SynchronousMachine.qPercent

11.1 Issue description

The description of `SynchronousMachine.qPercent` is “Percent of the coordinated reactive control that comes from this machine.”

Some tools treat the attribute differently. For some tools, the sum of `qPercent` of all `SynchronousMachines` regulating a node equals 100% for other this value could be higher than 100% (e.g. 5 generators all at 100% at one node).

11.2 Required implementation

The issue was discussed with WG13 on 9 March 2016. The conclusion is that the attribute `SynchronousMachine.qPercent` should be used as a participation factor not necessarily summing up to 100% for the participating devices in the control.

The reason for this statement is that the attribute is exchanged in EQ. The participation of the devices regulating particular node can be set in other profiles such as SSH. Therefore, the implementations should consider this in the load flow calculation and normalize the participation based on what devices are participating in the regulation of a particular node.