WG13 Issues - CIM Issues #4941

Equipment.aggregate on GeneratingUnit only to avoid redundancy

09/14/2021 03:47 PM - Herbert Falk

Status:	Closed		
Priority:	Low		
Target version:			
Author/Contact Info:	ENTSO-E N6	**Standard(s):**	
Base Release:	61970cim16v21	**Version:**	
Solution to be Applied To:		**Clause:**	
Solution Version:		**Sub-Clause:**	
Solution Applied By:		**Paragraph:**	
Completion Date:		**Table:**	
CIM Keywords:		**Originally Closed in Version:**	
Breaking Change:	No	**Origination Date:**	05/29/2013
Breaking Change Description:		**Origination ID:**	13_161
CIM Impacted Groups:	WG13	**Originally Assigned To:**	
Requestor:			

Description

Equipment.aggregate on GeneratingUnit
only to avoid redundant data at SynchronousMachine.
IOP agree to open WG13 discussion on this issue. What may need to be changed is to have change in UML that will forbid having multiple SynchronousMachines associated with one GeneratingUnit.

Proposed Solution

Move the aggregate attribute to ConductingEquipment.

Decision

It was decided not to move aggregated to ConductingEquipment.
Related issue 13_98.
Svein will start on a white paper, Jay and Lars-Ola will contribute.
2013-06-05/LOO

Oslo, 14-Jun-2023:
No change in CIM18. Constraints on profile level will need to restrict the usage of the attribute.

11-Oct-2023:
We don't see the need add the constraint which overrides the Oslo decision above. We have decided to close the ticket and leave as is.

History

#1 - 02/07/2023 01:39 PM - Chavdar Ivanov

- Subject changed from Equipment.aggregate on GeneratingUnit
only to avoid redunda to Equipment.aggregate on GeneratingUnitonly to avoid redunda

- Status changed from Open to Review

- Decision updated

WG13 to discuss and agree what we do here. We need to consider potential impact on teh usage of aggregate.

#2 - 06/14/2023 09:14 AM - Chavdar Ivanov

- Decision updated

#3 - 10/10/2023 06:52 AM - Todd Viegut

- Subject changed from Equipment.aggregate on GeneratingUnitonly to avoid redunda to Equipment.aggregate on GeneratingUnit only to avoid redundancy

#4 - 10/10/2023 06:54 AM - Todd Viegut

- Decision updated

#5 - 10/10/2023 06:57 AM - Todd Viegut

We have defined that a constraint needs to be written but the decision isn't specifying where. Should it be constrained to the GeneratingUnit (via inheritance from Equipment) or vice versa on SynchronousMachine and then have the constraint prohibit it on the GeneratingUnit. This needs to be clarified in the **Decision** notes.

#6 - 10/11/2023 10:36 AM - Todd Viegut

- Decision updated

#7 - 10/11/2023 10:37 AM - Todd Viegut

- Status changed from Review to Closed