



European Network of
Transmission System Operators
for Electricity

REMEDIAL ACTION PROFILE SPECIFICATION

2025-09-11

ICTC APPROVED
VERSION 2.4.0

Copyright notice:

Copyright © ENTSO-E. All Rights Reserved.

This document and its whole translations may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, except for literal and whole translation into languages other than English and under all circumstances, the copyright notice or references to ENTSO-E may not be removed.

This document and the information contained herein is provided on an "as is" basis.

ENTSO-E DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

This document is maintained by the ENTSO-E CIM WG. Comments or remarks are to be provided at cim@entsoe.eu

NOTE CONCERNING WORDING USED IN THIS DOCUMENT

The force of the following words is modified by the requirement level of the document in which they are used.

- **SHALL:** This word, or the terms "REQUIRED" or "MUST", means that the definition is an absolute requirement of the specification.
- **SHALL NOT:** This phrase, or the phrase "MUST NOT", means that the definition is an absolute prohibition of the specification.
- **SHOULD:** This word, or the adjective "RECOMMENDED", means that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
- **SHOULD NOT:** This phrase, or the phrase "NOT RECOMMENDED", means that there may exist valid reasons in particular circumstances when the particular behaviour is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behaviour described with this label.
- **MAY:** This word, or the adjective "OPTIONAL", means that an item is truly optional.

Revision History

Version	Date	Paragraph	Comments
1.0.0	2021-10-12		SOC approved.
2.0.0	2022-02-16		SOC approved.
2.1.0	2022-09-21		SOC approved.
2.2.0	2023-03-24		For review.
2.2.0	2023-04-20		For ICTC approval.
2.3.0-alpha	2024-02-17		For internal review.
2.3.0-beta	2024-03-20		For CIM WG review.
2.3.0-gamma	2024-04-09		For StG Strategy review.
2.3.1-alpha	2024-09-07		For CIM WG review.
2.3.2-alpha	2024-01-04		For CIM WG review.
2.4.0	2025-09-11		ICTC approval.

34 CONTENTS

35	Copyright notice:.....	2
36	Revision History.....	3
37	CONTENTS	4
38	1 Introduction	11
39	2 Application profile specification	11
40	2.1 Version information	11
41	2.2 Constraints naming convention	11
42	2.3 Profile constraints	12
43	2.4 Metadata.....	16
44	2.4.1 Constraints	16
45	2.4.2 Reference metadata	16
46	3 Detailed Profile Specification	16
47	3.1 General.....	16
48	3.2 (abstract) ACDCConverter root class	23
49	3.3 (NC) ACDCConverterAction	23
50	3.4 (abstract) ACDCTerminal	24
51	3.5 (NC) AvailabilityRemedialAction.....	24
52	3.6 (abstract) BatteryUnit root class	25
53	3.7 (NC) BatteryUnitAction.....	25
54	3.8 (abstract,NC) BiddingZone root class	25
55	3.9 (NC) CountertradeRemedialAction	25
56	3.10 (abstract) Contingency root class	26
57	3.11 (NC) ContingencyWithRemedialAction root class	26
58	3.12 (NC) ControlFunctionBlockAction	27
59	3.13 (abstract) DCTerminal root class	28
60	3.14 (abstract) EnergyConsumer root class	28
61	3.15 (abstract) EnergySource root class	28
62	3.16 (NC) EnergySourceModification	28
63	3.17 (abstract) EquivalentInjection root class	29
64	3.18 (NC) EquivalentInjectionAction.....	29
65	3.19 (abstract) ExternalNetworkInjection root class.....	29
66	3.20 (NC) ExternalNetworkInjectionAction	29
67	3.21 (NC) Gate	30
68	3.22 (abstract,NC) GateInputPin	30
69	3.23 (abstract,NC) GridStateAlteration	31
70	3.24 (NC) GridStateAlterationCollection	32
71	3.25 (NC) GridStateAlterationRemedialAction	32
72	3.26 (abstract) IdentifiedObject root class	33
73	3.27 (NC) IntertemporalPropertyRange	33
74	3.28 (NC) LoadAction	34
75	3.29 (abstract) Measurement root class	35
76	3.30 (NC) MeasurementCalculator	35
77	3.31 (NC) MeasurementCalculatorInput	35

78	3.32	(NC) PinContingency	36
79	3.33	(NC) PinDCTerminal	37
80	3.34	(NC) PinEquipment	37
81	3.35	(NC) PinGate	38
82	3.36	(NC) PinMeasurement	38
83	3.37	(NC) PinPowerTransferCorridor	39
84	3.38	(NC) PinTerminal	40
85	3.39	(NC) PinOperationalLimit	40
86	3.40	(abstract) PowerElectronicsConnection root class	41
87	3.41	(NC) PowerElectronicsConnectionAction	41
88	3.42	(abstract) PowerSystemResource	42
89	3.43	(abstract,NC) PowerTransferCorridor root class	42
90	3.44	(abstract,NC) PropertyReference root class	42
91	3.45	(NC) PTCActivePowerSupport root class	42
92	3.46	(NC) RedispatchRemedialAction	43
93	3.47	(abstract,NC) Region root class	43
94	3.48	(abstract) RegulatingControl root class	43
95	3.49	(NC) RegulatingControlAction	44
96	3.50	(abstract,NC) RemedialAction	44
97	3.51	(NC) RemedialActionScheme	46
98	3.52	(abstract) RotatingMachine root class	46
99	3.53	(NC) RotatingMachineAction	46
100	3.54	(NC) SchemeRemedialAction	47
101	3.55	(abstract,NC) SetPointAction	48
102	3.56	(abstract) ShuntCompensator root class	48
103	3.57	(NC) ShuntCompensatorModification	48
104	3.58	(NC) Stage	49
105	3.59	(NC) StageTrigger	49
106	3.60	(NC) StaticPropertyRange	50
107	3.61	(abstract) StaticVarCompensator root class	51
108	3.62	(NC) StaticVarCompensatorAction	51
109	3.63	(Description) Substation root class	52
110	3.64	(abstract) Switch root class	52
111	3.65	(abstract,NC) SystemOperator root class	52
112	3.66	(abstract) TapChanger root class	52
113	3.67	(NC) TapPositionAction	52
114	3.68	(abstract) Terminal root class	52
115	3.69	(NC) TopologyAction	53
116	3.70	(deprecated,NC) TriggerCondition	53
117	3.71	(NC) CalculationKind enumeration	54
118	3.72	Currency enumeration	54
119	3.73	(NC) GateLogicKind enumeration	58
120	3.74	(NC) LogicalOperatorsKind enumeration	58
121	3.75	(NC) PinDCTerminalKind enumeration	59
122	3.76	(NC) PinPowerTransferCorridorKind enumeration	59
123	3.77	(NC) RelativeDirectionKind enumeration	59

124	3.78	(NC) RemedialActionKind enumeration	59
125	3.79	(NC) RemedialActionSchemeKind enumeration	60
126	3.80	UnitSymbol enumeration	60
127	3.81	UnitMultiplier enumeration	61
128	3.82	(NC) ValueOffsetKind enumeration	61
129	3.83	ActivePower datatype	62
130	3.84	PerCent datatype	62
131	3.85	ReactivePower datatype	62
132	3.86	Seconds datatype	63
133	3.87	Boolean primitive	63
134	3.88	Duration primitive	63
135	3.89	Float primitive	63
136	3.90	Integer primitive	63
137	3.91	String primitive	63
138	3.92	(NC) BiddingZoneAction	63
139	3.93	(abstract,NC) ControlFunctionBlock root class	64
140	3.94	(abstract,NC) RangeConstraint	64
141	3.95	(NC) PinEquipmentTripping	65
142	3.96	(abstract) Equipment root class	65
143	3.97	(NC) RemedialActionDependency root class	65
144	3.98	(abstract,NC) FunctionInputVariable	66
145	3.99	(NC) RemedialActionDependencyKind enumeration	66
146	3.100	(abstract,NC) PowerRemedialAction	67
147	3.101	(NC) EquipmentControllerAction	68
148	3.102	(abstract,NC) EquipmentController root class	68
149	3.103	(NC) RemedialActionGroup	68
150	3.104	(NC) ElementCombinationConstraintKind enumeration	68
151	3.105	(NC,Description) PowerShiftKeyStrategy	69
152	3.106	(abstract) OperationalLimit root class	69
153	3.107	(abstract,NC) BiddingZoneBorder root class	69
154	3.108	(NC) PinTerminalKind enumeration	69
155	3.109	(NC) InServiceAction	70
156		Annex A (informative): Sample data	71
157	A.1	General	71
158	A.2	Sample instance data	71
159			

160 List of figures

161	Figure 1 – Class diagram RemedialActionProfile::Core	17
162	Figure 2 – Class diagram RemedialActionProfile::GridStateAlterationPart1	18
163	Figure 3 – Class diagram RemedialActionProfile::GridStateAlterationPart2	19
164	Figure 4 – Class diagram RemedialActionProfile::IntensityAndRange	20
165	Figure 5 – Class diagram RemedialActionProfile::RemedialAction	21
166	Figure 6 – Class diagram RemedialActionProfile::RemedialActionScheme	21
167	Figure 7 – Class diagram RemedialActionProfile::RemedialActionSchemeGate1	22

168	Figure 8 – Class diagram RemedialActionProfile::RemedialActionSchemeGate2	23
169		
170	List of tables	
171	Table 1 – Attributes of RemedialActionProfile::ACDCConverterAction	23
172	Table 2 – Association ends of RemedialActionProfile::ACDCConverterAction with other	
173	classes	24
174	Table 3 – Attributes of RemedialActionProfile::ACDCTerminal	24
175	Table 4 – Attributes of RemedialActionProfile::AvailabilityRemedialAction	24
176	Table 5 – Association ends of RemedialActionProfile::AvailabilityRemedialAction with	
177	other classes	25
178	Table 6 – Attributes of RemedialActionProfile::BatteryUnitAction	25
179	Table 7 – Association ends of RemedialActionProfile::BatteryUnitAction with other	
180	classes	25
181	Table 8 – Attributes of RemedialActionProfile::CountertradeRemedialAction	26
182	Table 9 – Association ends of RemedialActionProfile::CountertradeRemedialAction	
183	with other classes	26
184	Table 10 – Attributes of RemedialActionProfile::ContingencyWithRemedialAction	27
185	Table 11 – Association ends of	
186	RemedialActionProfile::ContingencyWithRemedialAction with other classes	27
187	Table 12 – Attributes of RemedialActionProfile::ControlFunctionBlockAction	27
188	Table 13 – Association ends of RemedialActionProfile::ControlFunctionBlockAction	
189	with other classes	28
190	Table 14 – Attributes of RemedialActionProfile::EnergySourceModification	28
191	Table 15 – Association ends of RemedialActionProfile::EnergySourceModification with	
192	other classes	28
193	Table 16 – Attributes of RemedialActionProfile::EquivalentInjectionAction	29
194	Table 17 – Association ends of RemedialActionProfile::EquivalentInjectionAction with	
195	other classes	29
196	Table 18 – Attributes of RemedialActionProfile::ExternalNetworkInjectionAction	30
197	Table 19 – Association ends of RemedialActionProfile::ExternalNetworkInjectionAction	
198	with other classes	30
199	Table 20 – Attributes of RemedialActionProfile::Gate	30
200	Table 21 – Attributes of RemedialActionProfile::GateInputPin	30
201	Table 22 – Association ends of RemedialActionProfile::GateInputPin with other classes	31
202	Table 23 – Attributes of RemedialActionProfile::GridStateAlteration	31
203	Table 24 – Association ends of RemedialActionProfile::GridStateAlteration with other	
204	classes	32
205	Table 25 – Attributes of RemedialActionProfile::GridStateAlterationCollection	32
206	Table 26 – Attributes of RemedialActionProfile::GridStateAlterationRemedialAction	32
207	Table 27 – Association ends of	
208	RemedialActionProfile::GridStateAlterationRemedialAction with other classes	33
209	Table 28 – Attributes of RemedialActionProfile::IdentifiedObject	33
210	Table 29 – Attributes of RemedialActionProfile::IntertemporalPropertyRange	34

211	Table 30 – Association ends of RemedialActionProfile::IntertemporalPropertyRange	
212	with other classes	34
213	Table 31 – Attributes of RemedialActionProfile::LoadAction	34
214	Table 32 – Association ends of RemedialActionProfile::LoadAction with other classes	34
215	Table 33 – Attributes of RemedialActionProfile::MeasurementCalculator	35
216	Table 34 – Attributes of RemedialActionProfile::MeasurementCalculatorInput	35
217	Table 35 – Association ends of RemedialActionProfile::MeasurementCalculatorInput	
218	with other classes	36
219	Table 36 – Attributes of RemedialActionProfile::PinContingency	36
220	Table 37 – Association ends of RemedialActionProfile::PinContingency with other	
221	classes	36
222	Table 38 – Attributes of RemedialActionProfile::PinDCTerminal	37
223	Table 39 – Association ends of RemedialActionProfile::PinDCTerminal with other	
224	classes	37
225	Table 40 – Attributes of RemedialActionProfile::PinEquipment	37
226	Table 41 – Association ends of RemedialActionProfile::PinEquipment with other	
227	classes	38
228	Table 42 – Attributes of RemedialActionProfile::PinGate	38
229	Table 43 – Association ends of RemedialActionProfile::PinGate with other classes	38
230	Table 44 – Attributes of RemedialActionProfile::PinMeasurement	39
231	Table 45 – Association ends of RemedialActionProfile::PinMeasurement with other	
232	classes	39
233	Table 46 – Attributes of RemedialActionProfile::PinPowerTransferCorridor	39
234	Table 47 – Association ends of RemedialActionProfile::PinPowerTransferCorridor with	
235	other classes	40
236	Table 48 – Attributes of RemedialActionProfile::PinTerminal	40
237	Table 49 – Association ends of RemedialActionProfile::PinTerminal with other classes	40
238	Table 50 – Attributes of RemedialActionProfile::PinOperationalLimit	40
239	Table 51 – Association ends of RemedialActionProfile::PinOperationalLimit with other	
240	classes	41
241	Table 52 – Attributes of RemedialActionProfile::PowerElectronicsConnectionAction	41
242	Table 53 – Association ends of	
243	RemedialActionProfile::PowerElectronicsConnectionAction with other classes	41
244	Table 54 – Attributes of RemedialActionProfile::PowerSystemResource	42
245	Table 55 – Attributes of RemedialActionProfile::PTCActivePowerSupport	42
246	Table 56 – Association ends of RemedialActionProfile::PTCActivePowerSupport with	
247	other classes	42
248	Table 57 – Attributes of RemedialActionProfile::RedispatchRemedialAction	43
249	Table 58 – Association ends of RemedialActionProfile::RedispatchRemedialAction with	
250	other classes	43
251	Table 59 – Attributes of RemedialActionProfile::RegulatingControlAction	44
252	Table 60 – Association ends of RemedialActionProfile::RegulatingControlAction with	
253	other classes	44
254	Table 61 – Attributes of RemedialActionProfile::RemedialAction	45

255	Table 62 – Association ends of RemedialActionProfile::RemedialAction with other	
256	classes	46
257	Table 63 – Attributes of RemedialActionProfile::RemedialActionScheme	46
258	Table 64 – Association ends of RemedialActionProfile::RemedialActionScheme with	
259	other classes	46
260	Table 65 – Attributes of RemedialActionProfile::RotatingMachineAction	47
261	Table 66 – Association ends of RemedialActionProfile::RotatingMachineAction with	
262	other classes	47
263	Table 67 – Attributes of RemedialActionProfile::SchemeRemedialAction	47
264	Table 68 – Association ends of RemedialActionProfile::SchemeRemedialAction with	
265	other classes	48
266	Table 69 – Attributes of RemedialActionProfile::SetPointAction	48
267	Table 70 – Association ends of RemedialActionProfile::SetPointAction with other	
268	classes	48
269	Table 71 – Attributes of RemedialActionProfile::ShuntCompensatorModification	48
270	Table 72 – Association ends of RemedialActionProfile::ShuntCompensatorModification	
271	with other classes	49
272	Table 73 – Attributes of RemedialActionProfile::Stage	49
273	Table 74 – Association ends of RemedialActionProfile::Stage with other classes	49
274	Table 75 – Attributes of RemedialActionProfile::StageTrigger	50
275	Table 76 – Association ends of RemedialActionProfile::StageTrigger with other classes	50
276	Table 77 – Attributes of RemedialActionProfile::StaticPropertyRange	50
277	Table 78 – Association ends of RemedialActionProfile::StaticPropertyRange with other	
278	classes	51
279	Table 79 – Attributes of RemedialActionProfile::StaticVarCompensatorAction	51
280	Table 80 – Association ends of RemedialActionProfile::StaticVarCompensatorAction	
281	with other classes	51
282	Table 81 – Attributes of RemedialActionProfile::TapPositionAction	52
283	Table 82 – Association ends of RemedialActionProfile::TapPositionAction with other	
284	classes	52
285	Table 83 – Attributes of RemedialActionProfile::TopologyAction	53
286	Table 84 – Association ends of RemedialActionProfile::TopologyAction with other	
287	classes	53
288	Table 85 – Attributes of RemedialActionProfile::TriggerCondition	53
289	Table 86 – Association ends of RemedialActionProfile::TriggerCondition with other	
290	classes	53
291	Table 87 – Literals of RemedialActionProfile::CalculationKind	54
292	Table 88 – Literals of RemedialActionProfile::Currency	54
293	Table 89 – Literals of RemedialActionProfile::GateLogicKind	58
294	Table 90 – Literals of RemedialActionProfile::LogicalOperatorsKind	59
295	Table 91 – Literals of RemedialActionProfile::PinDCTerminalKind	59
296	Table 92 – Literals of RemedialActionProfile::PinPowerTransferCorridorKind	59
297	Table 93 – Literals of RemedialActionProfile::RelativeDirectionKind	59

298	Table 94 – Literals of RemedialActionProfile::RemedialActionKind	60
299	Table 95 – Literals of RemedialActionProfile::RemedialActionSchemeKind	60
300	Table 96 – Literals of RemedialActionProfile::UnitSymbol	61
301	Table 97 – Literals of RemedialActionProfile::UnitMultiplier	61
302	Table 98 – Literals of RemedialActionProfile::ValueOffsetKind	62
303	Table 99 – Attributes of RemedialActionProfile::ActivePower	62
304	Table 100 – Attributes of RemedialActionProfile::PerCent	62
305	Table 101 – Attributes of RemedialActionProfile::ReactivePower	62
306	Table 102 – Attributes of RemedialActionProfile::Seconds	63
307	Table 103 – Attributes of RemedialActionProfile::BiddingZoneAction	63
308	Table 104 – Association ends of RemedialActionProfile::BiddingZoneAction with other	
309	classes	64
310	Table 105 – Attributes of RemedialActionProfile::RangeConstraint	64
311	Table 106 – Association ends of RemedialActionProfile::RangeConstraint with other	
312	classes	64
313	Table 107 – Attributes of RemedialActionProfile::PinEquipmentTripping	65
314	Table 108 – Association ends of RemedialActionProfile::PinEquipmentTripping with	
315	other classes	65
316	Table 109 – Attributes of RemedialActionProfile::RemedialActionDependency	66
317	Table 110 – Association ends of RemedialActionProfile::RemedialActionDependency	
318	with other classes	66
319	Table 111 – Attributes of RemedialActionProfile::FunctionInputVariable	66
320	Table 112 – Literals of RemedialActionProfile::RemedialActionDependencyKind	66
321	Table 113 – Attributes of RemedialActionProfile::PowerRemedialAction	67
322	Table 114 – Association ends of RemedialActionProfile::PowerRemedialAction with	
323	other classes	67
324	Table 115 – Attributes of RemedialActionProfile::EquipmentControllerAction	68
325	Table 116 – Association ends of RemedialActionProfile::EquipmentControllerAction	
326	with other classes	68
327	Table 117 – Attributes of RemedialActionProfile::RemedialActionGroup	68
328	Table 118 – Literals of RemedialActionProfile::ElementCombinationConstraintKind	69
329	Table 119 – Attributes of RemedialActionProfile::PowerShiftKeyStrategy	69
330	Table 120 – Association ends of RemedialActionProfile::PowerShiftKeyStrategy with	
331	other classes	69
332	Table 121 – Literals of RemedialActionProfile::PinTerminalKind	70
333	Table 122 – Attributes of RemedialActionProfile::InServiceAction	70
334	Table 123 – Association ends of RemedialActionProfile::InServiceAction with other	
335	classes	70
336		

1 Introduction

The remedial action profile enables an exchange of the remedial actions for the purpose of NC related business processes.

A remedial action means any measure applied by a TSO or several TSOs, manually or automatically, in order to maintain operational security.¹

An available remedial action is a remedial action which is available to solve identified constraints. It includes the needed technical and cost information.²

The available remedial actions are input data for security analysis.

The available remedial action profile enables the exchange of both curative and preventive remedial actions. Grid state alterations (the change in the power system state that should be applied) are defined for each remedial action. The definition of grid state alterations allows for constraining or further precisising some of the properties available in the IGM. Grid state alterations can be configured for every parameter of the steady state hypothesis instance data from the IGM. The available remedial action profile provides information on the availability of the remedial actions. In cases where it is necessary to only update the status of the remedial action, only an instance of RemedialAction class can be exchanged without any other objects from the profile.

2 Application profile specification

2.1 Version information

The content is generated from UML model file CIM17-2_CGMES31v01_PROF-20v02_NC24v18_MS10v03_DES10v01.eap.

This edition is based on the IEC 61970 UML version 'IEC61970CIM17v40', dated '2020-08-24'.

- Title: Remedial action Vocabulary
- Keyword: RA
- Description: This vocabulary is describing the remedial action profile.
- Version IRI: <https://ap-voc.cim4.eu/RemedialAction/2.4>
- Version info: 2.4.0
- Prior version: <https://ap-voc.cim4.eu/RemedialAction/2.3>
- Conforms to: urn:iso:std:iec:61970-600-2:ed-1|urn:iso:std:iec:61970-301:ed-7:amd1|file:///iec61970cim17v40_iec61968cim13v13a_iec62325cim03v17a.eap|urn:iso:std:iec:61970-401:draft:ed-1|urn:iso:std:iec:61970-501:draft:ed-2|file:///CIM17-2_CGMES31v01_PROF-20v02_NC24v18_MS10v03_DES10v01.eap
- Identifier: urn:uuid:57fcfe0e-258c-45f2-b2ed-ff5b6a9859bc

2.2 Constraints naming convention

The naming of the rules shall not be used for machine processing. The rule names are just a string. The naming convention of the constraints is as follows.

"{rule.Type}:{rule.Standard}:{rule.Profile}:{rule.Property}:{rule.Name}"

¹ [SOURCE: CACM art.2.13]

² [SOURCE: 2019 Inter-RSC report]

375 where

376 rule.Type: C – for constraint; R – for requirement

377 rule.Standard: the number of the standard e.g. 301 for 61970-301, 456 for 61970-456, 13 for
378 61968-13. 61970-600 specific constraints refer to 600 although they are related to one or
379 combination of the 61970-450 series profiles. For NC profiles, NC is used.

380 rule.Profile: the abbreviation of the profile, e.g. TP for Topology profile. If set to “ALL” the
381 constraint is applicable to all IEC 61970-600 profiles.

382 rule.Property: for UML classes, the name of the class, for attributes and associations, the name
383 of the class and attribute or association end, e.g. EnergyConsumer, IdentifiedObject.name, etc.
384 If set to “NA” the property is not applicable to a specific UML element.

385 rule.Name: the name of the rule. It is unique for the same property.

386 Example: C:600:ALL:IdentifiedObject.name:stringLength

387 2.3 Profile constraints

388 This clause defines requirements and constraints that shall be fulfilled by applications that
389 conform to this document.

390 This document is the master for rules and constraints tagged "NC". For the sake of self-
391 containment, the list below also includes a copy of the relevant rules from IEC 61970-452,
392 tagged "452".

- 393 • C:452:ALL:NA:datatypes

394 According to 61970-501, datatypes are not exchanged in the instance data. The
395 UnitMultiplier is 1 in cases none value is specified in the profile.

- 396 • R:452:ALL:NA:exchange

397 Optional and required attributes and associations must be imported and exported if they
398 are in the model file prior to import.

- 399 • R:452:ALL:NA:exchange1

400 If an optional attribute does not exist in the imported file, it does not have to be exported
401 in case exactly the same data set is exported, i.e. the tool is not obliged to automatically
402 provide this attribute. If the export is resulting from an action by the user performed after
403 the import, e.g. data processing or model update the export can contain optional
404 attributes.

- 405 • R:452:ALL:NA:exchange2

406 In most of the profiles the selection of optional and required attributes is made so as to
407 ensure a minimum set of required attributes without which the exchange does not fulfil
408 its basic purpose. Business processes governing different exchanges can require
409 mandatory exchange of certain optional attributes or associations. Optional and required
410 attributes and associations shall therefore be supported by applications which claim
411 conformance with certain functionalities of the IEC 61970-452. This provides flexibility
412 for the business processes to adapt to different business requirements and base the
413 exchanges on IEC 61970-452 compliant applications.

- 414 • R:452:ALL:NA:exchange3

415 An exporter may, at his or her discretion, produce a serialization containing additional
416 class data described by the CIM Schema but not required by this document provided
417 these data adhere to the conventions established in Clause 5.

418 • R:452:ALL:NA:exchange4

419 From the standpoint of the model import used by a data recipient, the document
420 describes a subset of the CIM that importing software shall be able to interpret in order
421 to import exported models. Data providers are free to exceed the minimum requirements
422 described herein as long as their resulting data files are compliant with the CIM Schema
423 and the conventions established in Clause 5. The document, therefore, describes
424 additional classes and class data that, although not required, exporters will, in all
425 likelihood, choose to include in their data files. The additional classes and data are
426 labelled as required (cardinality 1..1) or as optional (cardinality 0..1) to distinguish them
427 from their required counterparts. Please note, however, that data importers could
428 potentially receive data containing instances of any and all classes described by the
429 CIM Schema.

430 • R:452:ALL:NA:cardinality

431 The cardinality defined in the CIM model shall be followed, unless a more restrictive
432 cardinality is explicitly defined in this document. For instance, the cardinality on the
433 association between VoltageLevel and BaseVoltage indicates that a VoltageLevel shall
434 be associated with one and only one BaseVoltage, but a BaseVoltage can be associated
435 with zero to many VoltageLevels.

436 • R:452:ALL:NA:associations

437 Associations between classes referenced in this document and classes not referenced
438 here are not required regardless of cardinality.

439 • R:452:ALL:IdentifiedObject.name:rule

440 The attribute “name” inherited by many classes from the abstract class IdentifiedObject
441 is not required to be unique. It must be a human readable identifier without additional
442 embedded information that would need to be parsed. The attribute is used for purposes
443 such as User Interface and data exchange debugging. The MRID defined in the data
444 exchange format is the only unique and persistent identifier used for this data exchange.
445 The attribute IdentifiedObject.name is, however, always required for CoreEquipment
446 profile and Short Circuit profile.

447 • R:452:ALL:IdentifiedObject.description:rule

448 The attribute “description” inherited by many classes from the abstract class
449 IdentifiedObject must contain human readable text without additional embedded
450 information that would need to be parsed.

451 • R:452:ALL:NA:uniqueIdentifier

452 All IdentifiedObject-s shall have a persistent and globally unique identifier (Master
453 Resource Identifier - mRID).

454 • R:452:ALL:NA:unitMultiplier

455 For exchange of attributes defined using CIM Data Types (ActivePower, Susceptance,
456 etc.) a unit multiplier of 1 is used if the UnitMultiplier specified in this document is “none”.

457 • C:452:ALL:IdentifiedObject.name:stringLength

- 458 The string IdentifiedObject.name has a maximum of 128 characters.
- 459 • C:452:ALL:IdentifiedObject.description:stringLength
- 460 The string IdentifiedObject.description is maximum 256 characters.
- 461 • C:452:ALL:NA:float
- 462 An attribute that is defined as float (e.g. has a type Float or a type which is a Datatype
463 with .value attribute of type Float) shall support ISO/IEC 60559:2020 for floating-point
464 arithmetic using single precision floating point. A single precision float supports 7
465 significant digits where the significant digits are described as an integer, or a decimal
466 number with 6 decimal digits. Two float values are equal when the significant with 7
467 digits are identical, e.g. 1234567 is equal 1.234567E6 and so are 1.2345678 and
468 1.234567E0.
- 469 • R:NC:ALL:Region:reference
- 470 The reference to the Region is normally a reference to the capacity calculation region,
471 which is identified by “Y” EIC code of the capacity calculation region.
- 472 • R:NC:ALL:SystemOperator:reference
- 473 The reference to the System Operator is normally identified by “X” EIC code of TSO.
- 474 • R:NC:ALL:NA:serialization
- 475 The profiles are defined in the EnterpriseArchitect application and have multiple artifacts
476 that describe them. The main artifacts are:
- 477 1) the EAP file (EnterpriseArchitect project file),
478 2) the profiles’ specification document and
479 3) the application profiles (RDFS and SHACL).
- 480 Due to the complexity of the profiles, there are various cross profile associations that,
481 from profiling and profile maintenance point of view, it is not practical to include the
482 complete inheritance structure in all profiles. If this is done the documentation provided
483 for all profiles would also include duplicated information on the description of classes
484 defined in other profiles. The following cases are often observed in profiles:
- 485 ○ Case 1: An association end refers to an abstract class
486 ○ Case 2: An abstract class (stereotyped with “Description”) has an association
487 (direction to another class)
488 ○ Case 3: An abstract class (not stereotyped with “Description”) has an
489 association (direction to another class)
490 ○ Case 4: An abstract class has attributes and subclasses are not in the profile
- 491 In all cases, the datasets shall only include the subtypes of the abstract classes with
492 the related properties (i.e. association or attributes) defined in the profile. The
493 information is taken from either canonical model or the profiles where complete
494 (expected) inheritance structure for the related abstract class is described. SHACL
495 based constraints include constraints only for the concrete classes that are subtypes of
496 the abstract class in the profile, and this can be used to inform which are the concrete
497 classes expected in a dataset that conforms to this profile.
- 498 It should be taken into account that this approach deviates from MVAL5 (IEC 61970-
499 600-1:2021), which creates multiple inheritance at serialization. For instance, with this
500 more explicit exchange the serialization of the association between abstract class
501 Equipment and abstract class Circuit for a PowerTransformer will be serialized as
502 follows:

503 ○ for association
504 <cim:PowerTransformer rdf:about="_c328f787-bc17-47ad-a59f-6ba7133340d0">
505 <nc:Equipment.Circuit rdf:resource="#_9ced16ac-d076-4ef9-a241-a998a579e77b"/>
506 </cim:PowerTransformer>

507 ○ for attribute
508 <cim:ACLineSegment rdf:about="_04f681aa-6999-4fb3-9775-acaa5eb7ceff">
509 <cim:Equipment.inService>true</cim:Equipment.inService>
510 </cim:ACLineSegment>

511 The usage of rdf:ID or rdf:about depends on the stereotype of the class. rdf:about is
512 used if the class has the stereotype "Description".

513 An example of not allowed serialization, as the Equipment is an abstract class

514 <cim:Equipment rdf:about="_c328f787-bc17-47ad-a59f-6ba7133340d0">
515 <nc:Equipment.Circuit rdf:resource="#_9ced16ac-d076-4ef9-a241-a998a579e77b"/>
516 </cim:Equipment>

517 • C:NC:RA:GridStateAlteration:associations

518 A GridStateAlteration shall have either association to
519 GridStateAlterationRemedialAction or GridStateAlterationCollection.

520 • C:NC:RA:StaticPropertyRange.PropertyReference:value

521 The value of the association end StaticPropertyRange.PropertyReference shall be one
522 of the values published in the skos:ConceptScheme
523 <https://energy.referencedata.eu/PropertyReference/>.

524 • C:NC:RA:GridStateAlteration.PropertyReference:value

525 The value of the association end GridStateAlteration.PropertyReference shall be one of
526 the values published in the skos:ConceptScheme
527 <https://energy.referencedata.eu/PropertyReference/>.

528 • C:NC:RA:PinEquipment.PropertyReference:value

529 The value of the association end PinEquipment.PropertyReference shall be one of the
530 values published in the skos:ConceptScheme
531 <https://energy.referencedata.eu/PropertyReference/>.

532 • C:NC:RA:GateInputPin.logicKind:required

533 The attribute is required for the following classes: PinMeasurement,
534 PinOperationalLimit, PinPowerTransferCorridor and PinDCTerminal.

535

536

2.4 Metadata

ENTSO-E agreed to extend the header and metadata definitions by IEC 61970-552 Ed2. This new header definitions rely on W3C recommendations which are used worldwide and are positively recognised by the European Commission. The new definitions of the header mainly use Provenance ontology (PROV-O), Time Ontology and Data Catalog Vocabulary (DCAT). The global new header applicable for this profile is included in the metadata and document header specification document.

The header vocabulary contains all attributes defined in IEC 61970-552. This is done only for the purpose of having one vocabulary for header and to ensure transition for data exchanges that are using IEC 61970-552:2016 header. This profile does not use IEC 61970-552:2016 header attributes and relies only on the extended attributes.

2.4.1 Constraints

The identification of the constraints related to the metadata follows the same convention for naming of the constraints as for profile constraints.

- R:NC:ALL:wasAttributedTo:usage

The prov:wasAttributedTo should normally be the “X” EIC code of the actor or their URI (prov:Agent).

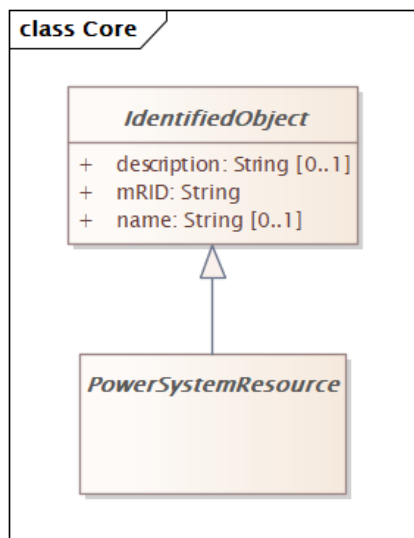
2.4.2 Reference metadata

The header defined for this profile requires availability of a set of reference metadata. For instance, the attribute prov:wasGeneratedBy requires a reference to an activity which produced the model or the related process. The activities are defined as reference metadata and their identifiers are referenced from the header to enable the receiving entity to retrieve the “static” (reference) information that is not modified frequently. This approach imposes a requirement that both the sending entity and the receiving entity have access to a unique version of the reference metadata. Therefore, each business process shall define which reference metadata is used and where it is located.

3 Detailed Profile Specification

3.1 General

This package contains remedial action profile.



567

568

Figure 1 – Class diagram RemedialActionProfile::Core

569

Figure 1: The diagram shows classes from Base CIM used in the profile.

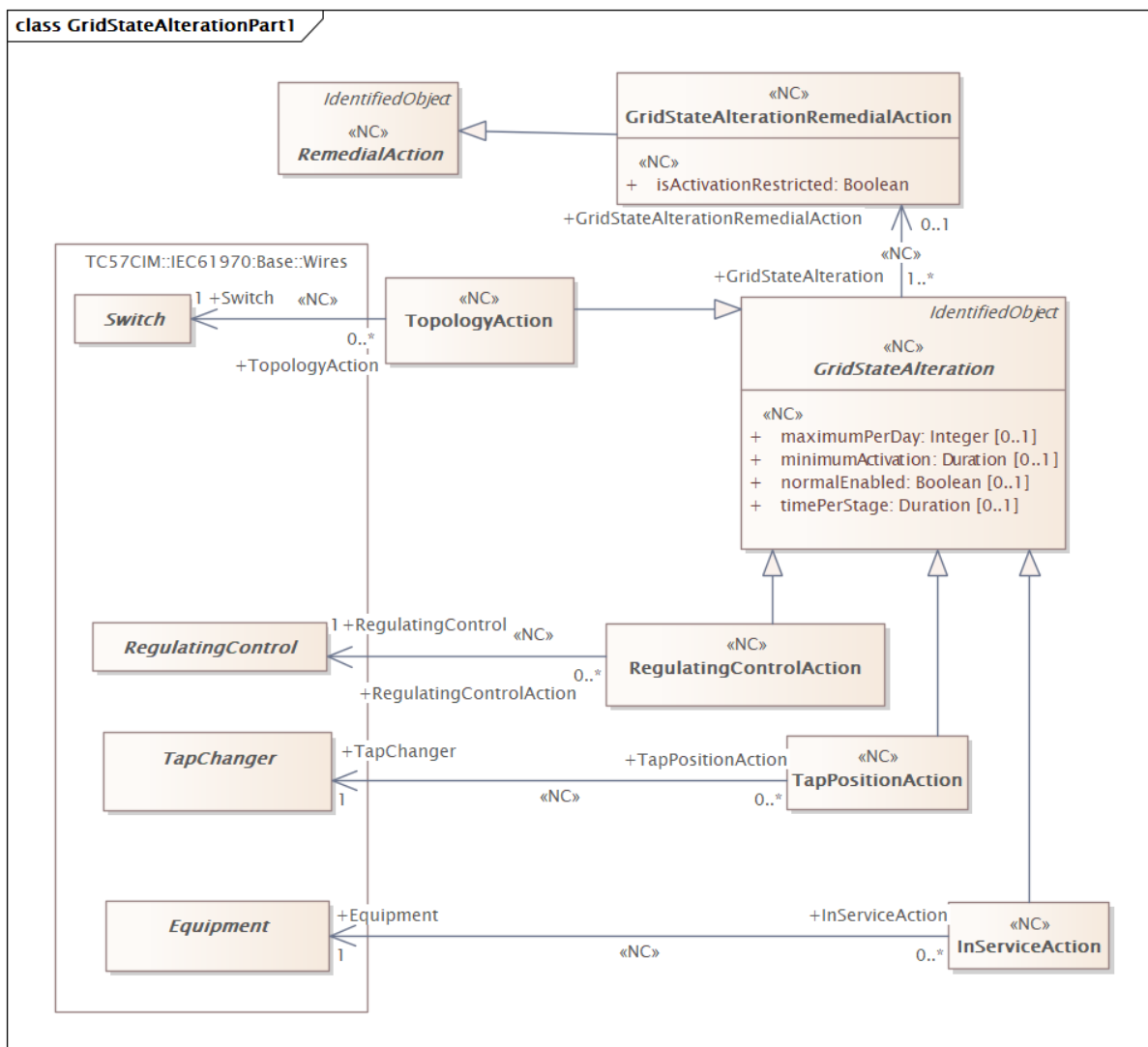


Figure 2 – Class diagram RemedialActionProfile::GridStateAlterationPart1

Figure 2: This diagram contains extended classes for the purpose of the remedial action data exchange.

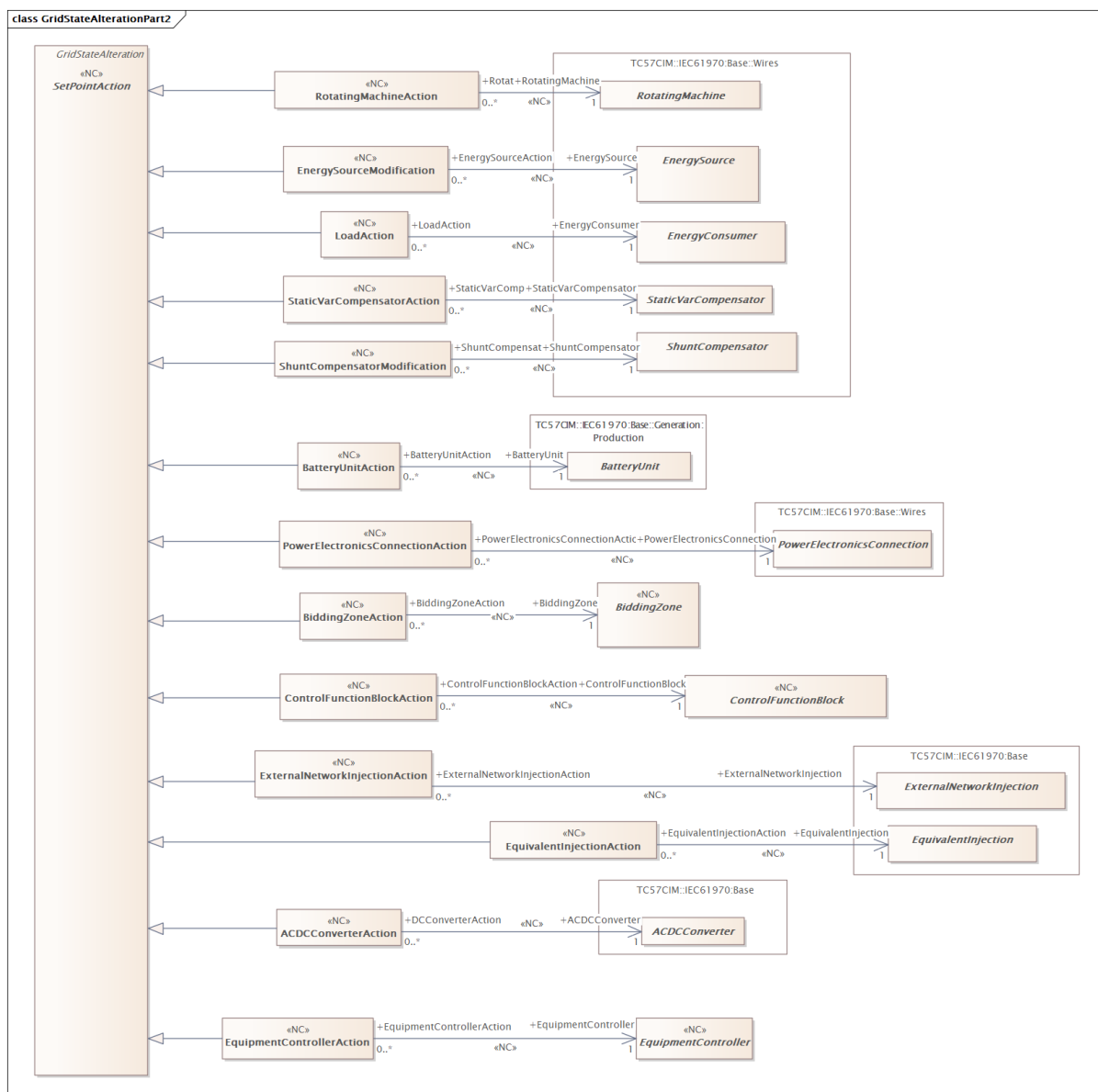


Figure 3 – Class diagram RemedialActionProfile::GridStateAlterationPart2

Figure 3: This diagram contains extended classes for the purpose of the remedial action data exchange.

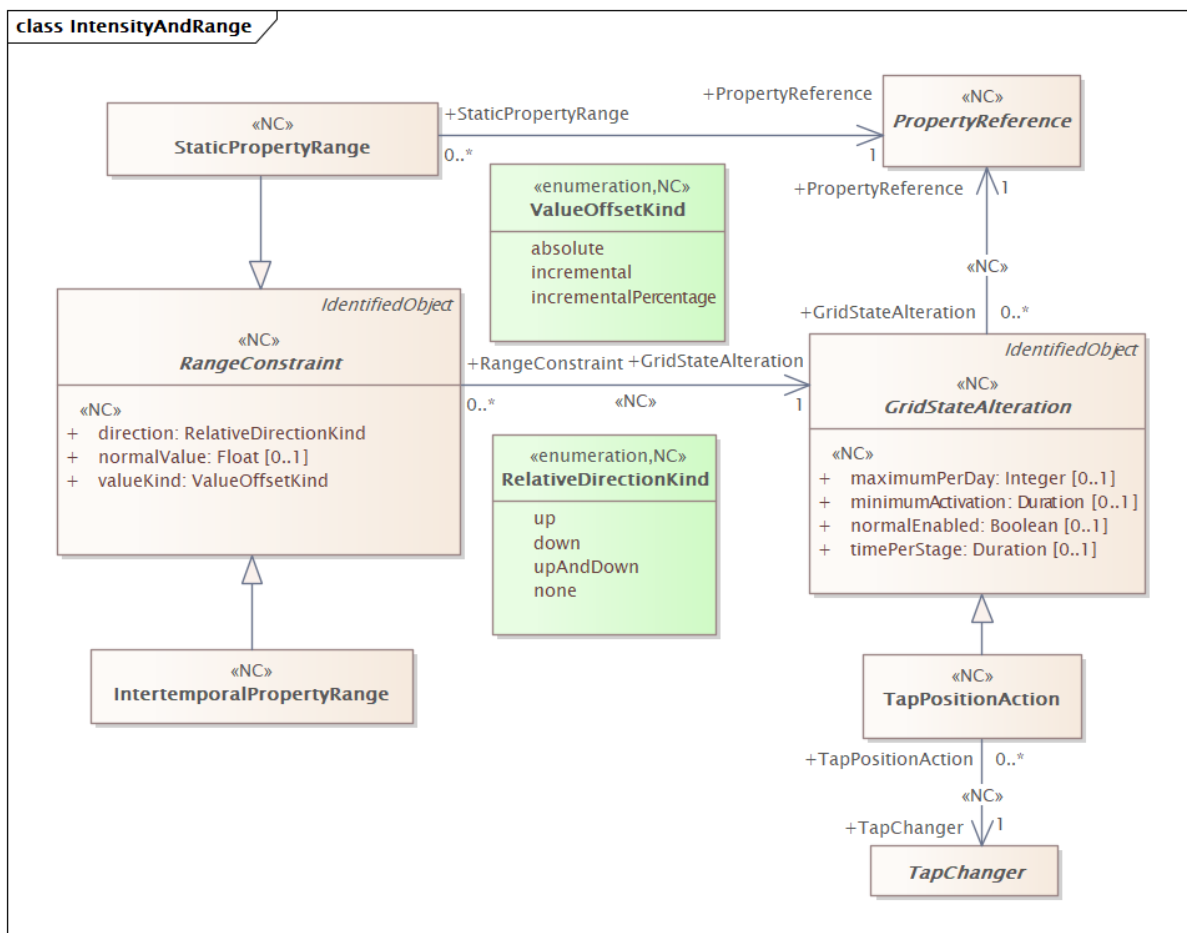


Figure 4 – Class diagram RemedialActionProfile::IntensityAndRange

Figure 4: This diagram contains extended classes related to the modelling of static, dynamic ranges and intensity.

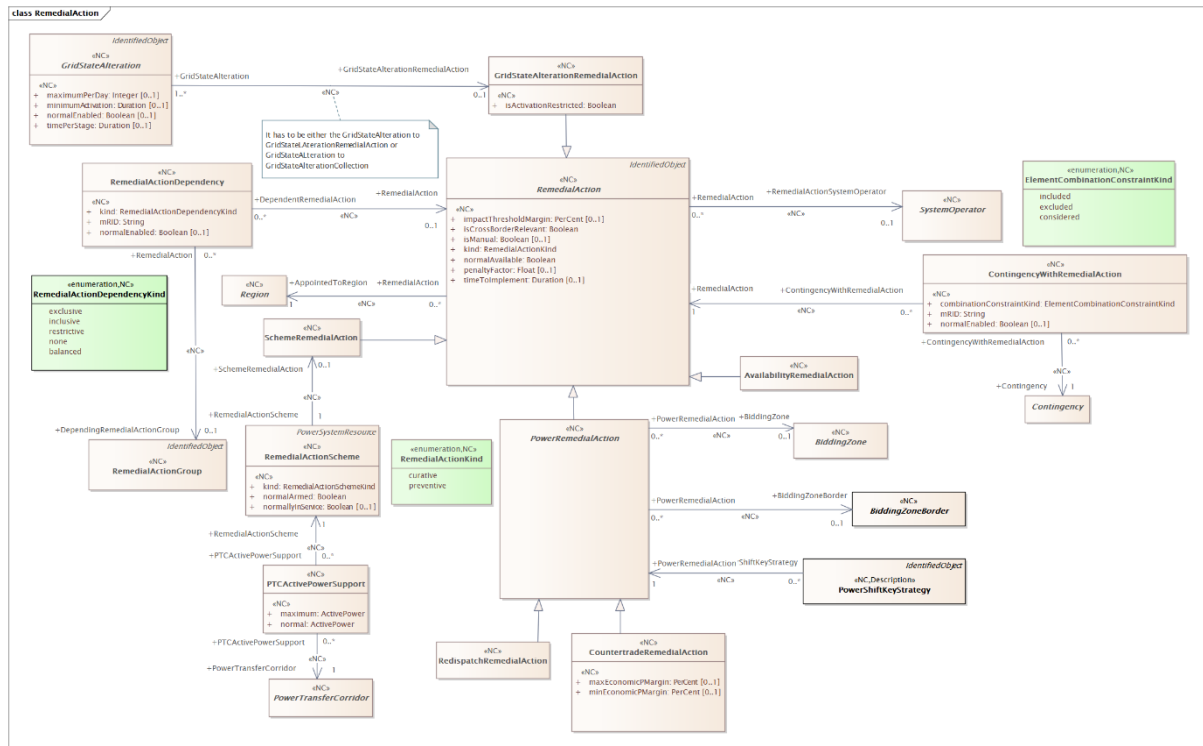


Figure 5 – Class diagram RemedialActionProfile::RemedialAction

Figure 5: The diagram contains main classes related to the remedial action.

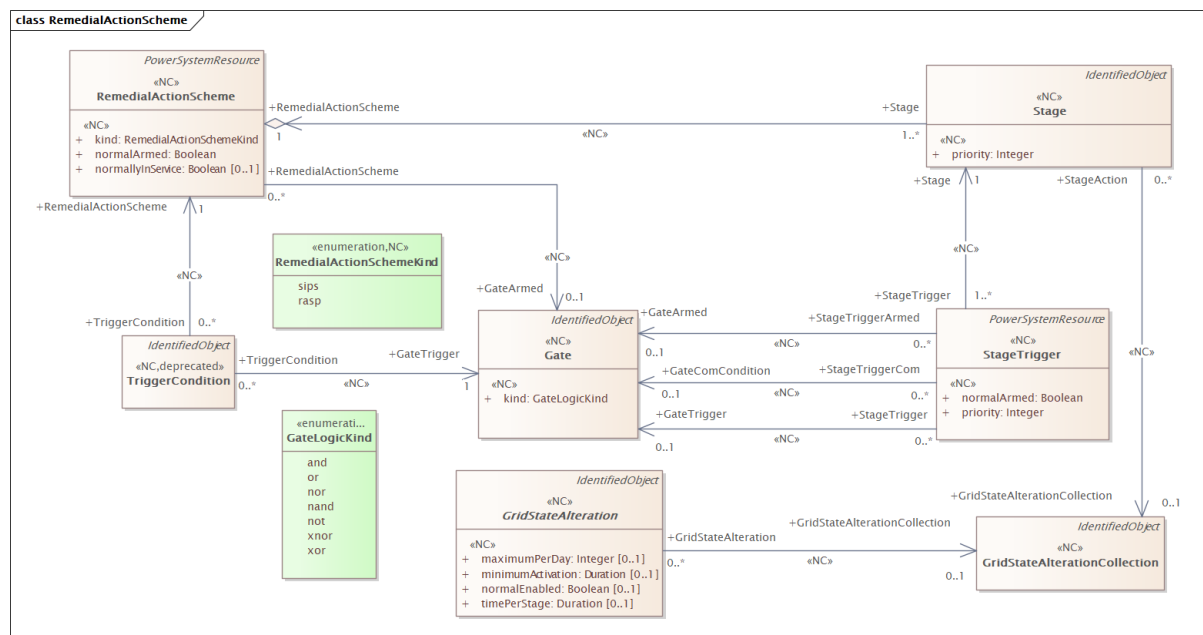


Figure 6 – Class diagram RemedialActionProfile::RemedialActionScheme

Figure 6: The diagram shows remedial action scheme related classes.

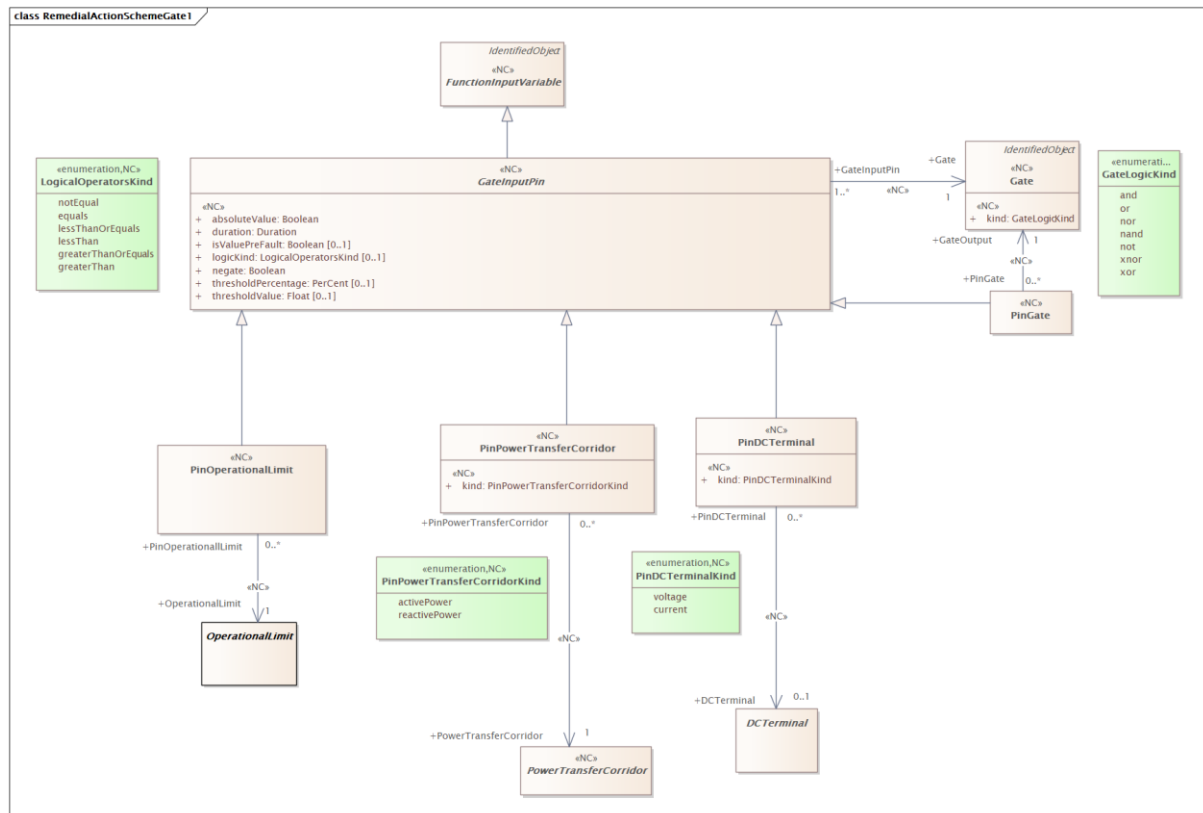


Figure 7 – Class diagram RemedialActionProfile::RemedialActionSchemeGate1

Figure 7: The diagram shows remedial action scheme gate related classes.

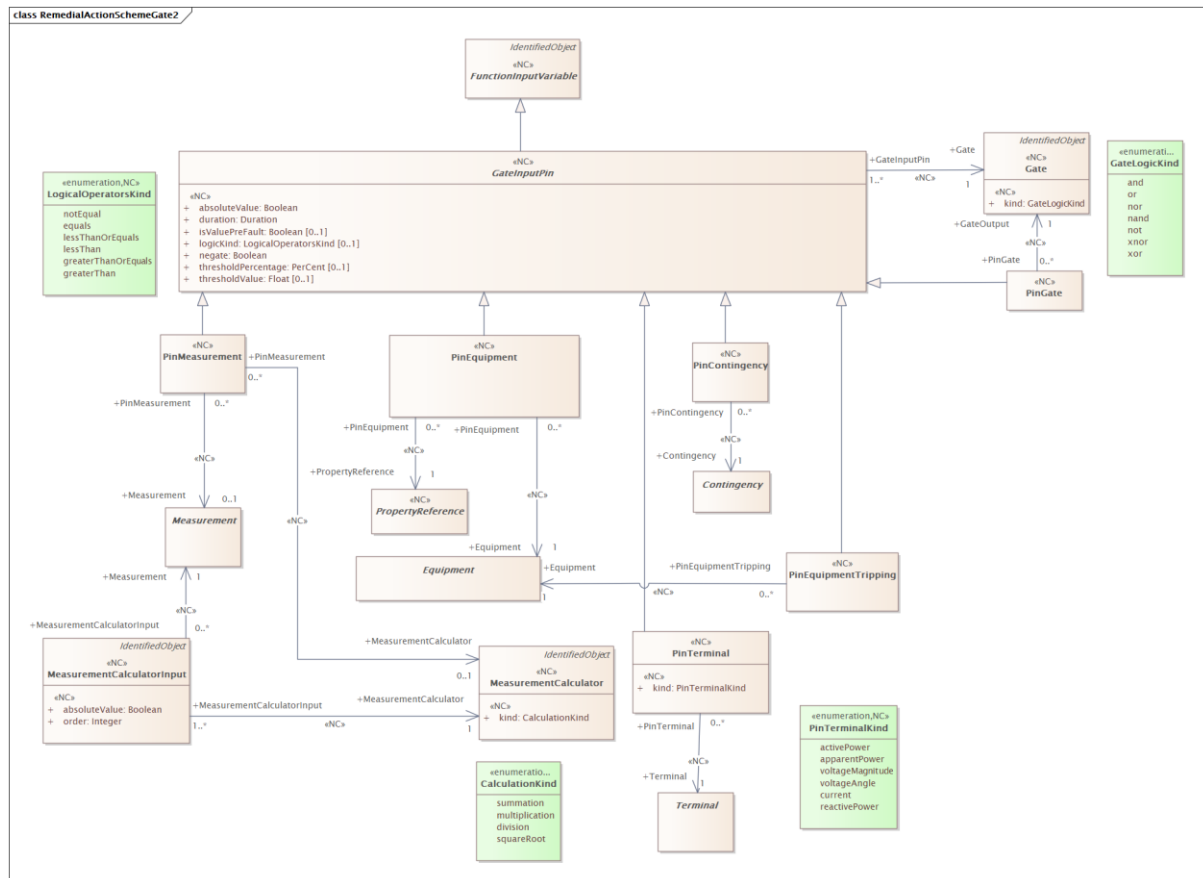


Figure 8 – Class diagram RemedialActionProfile::RemedialActionSchemeGate2

Figure 8: The diagram shows remedial action scheme gate related classes.

3.2 (abstract) ACDCCConverter root class

A unit with valves for three phases, together with unit control equipment, essential protective and switching devices, DC storage capacitors, phase reactors and auxiliaries, if any, used for conversion.

3.3 (NC) ACDCCConverterAction

Inheritance path = [SetPointAction](#) : [GridStateAlteration](#) : [IdentifiedObject](#)

Alternate current Direct current (ACDC) converter action.

Table 1 shows all attributes of ACDCCConverterAction.

Table 1 – Attributes of RemedialActionProfile::ACDCCConverterAction

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 2 shows all association ends of ACDCCConverterAction with other classes.

Table 2 – Association ends of RemedialActionProfile::ACDCConverterAction with other classes

mult from	name	mult to	type	description
0..*	ACDCConverter	1..1	ACDCConverter	(NC) The ACDCConverter that is associated with an action.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

3.4 (abstract) ACDCTerminal

Inheritance path = [IdentifiedObject](#)

An electrical connection point (AC or DC) to a piece of conducting equipment. Terminals are connected at physical connection points called connectivity nodes.

Table 3 shows all attributes of ACDCTerminal.

Table 3 – Attributes of RemedialActionProfile::ACDCTerminal

name	mult	type	description
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

3.5 (NC) AvailabilityRemedialAction

Inheritance path = [RemedialAction](#) : [IdentifiedObject](#)

Availability remedial action is a remedial action that cancels or reschedules an availability schedule.

Table 4 shows all attributes of AvailabilityRemedialAction.

Table 4 – Attributes of RemedialActionProfile::AvailabilityRemedialAction

name	mult	type	description
kind	1..1	RemedialActionKind	(NC) inherited from: RemedialAction
penaltyFactor	0..1	Float	(NC) inherited from: RemedialAction
isCrossBorderRelevant	1..1	Boolean	(NC) inherited from: RemedialAction
isManual	0..1	Boolean	(NC) inherited from: RemedialAction
timeToImplement	0..1	Duration	(NC) inherited from: RemedialAction
impactThresholdMargin	0..1	PerCent	(NC) inherited from: RemedialAction
normalAvailable	1..1	Boolean	(NC) inherited from: RemedialAction
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 5 shows all association ends of AvailabilityRemedialAction with other classes.

Table 5 – Association ends of RemedialActionProfile::AvailabilityRemedialAction with other classes

mult from	name	mult to	type	description
0..*	AppointedToRegion	1..1	Region	(NC) inherited from: RemedialAction
0..*	RemedialActionSystemOperator	0..1	SystemOperator	(NC) inherited from: RemedialAction

3.6 (abstract) BatteryUnit root class

An electrochemical energy storage device.

3.7 (NC) BatteryUnitAction

Inheritance path = [SetPointAction](#) : [GridStateAlteration](#) : [IdentifiedObject](#)

Battery unit setpoint action.

Table 6 shows all attributes of BatteryUnitAction.

Table 6 – Attributes of RemedialActionProfile::BatteryUnitAction

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 7 shows all association ends of BatteryUnitAction with other classes.

Table 7 – Association ends of RemedialActionProfile::BatteryUnitAction with other classes

mult from	name	mult to	type	description
0..*	BatteryUnit	1..1	BatteryUnit	(NC) The BatteryUnit that is associated with an action.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

3.8 (abstract,NC) BiddingZone root class

A bidding zone is a market-based method for handling power transmission congestion. It consists of scheduling areas that include the relevant production (supply) and consumption (demand) to form an electrical area with the same market price without capacity allocation.

3.9 (NC) CountertradeRemedialAction

Inheritance path = [PowerRemedialAction](#) : [RemedialAction](#) : [IdentifiedObject](#)

Countertrade is a remedial action to relieve physical congestions where the location of activated resources within the bidding zone is not known.

Table 8 shows all attributes of CountertradeRemedialAction.

647 **Table 8 – Attributes of RemedialActionProfile::CountertradeRemedialAction**

name	mult	type	description
maxEconomicPMargin	0..1	PerCent	(NC) High economic active power limit given by the percentage of the relevant units operating p. e.g. If a generating unit (G1) with maximum operating active power of 100 MW and a conform load with active maximum load of 50 MW (L1). Max economic p margin of 90% will give the limit of the shift key to be 90 MW for the G1 and 45 MW for the L1. The allowed value range is [0,100].
minEconomicPMargin	0..1	PerCent	(NC) Low economic active power limit given by the percentage of the relevant units operating p. e.g. If a generating unit (G1) with minimum operating active power of 10 MW and a conform load with active maximum load of 5 MW (L1). Min economic p margin of 90% will give the limit of the shift key to be 11 MW for the G1 and 5.5 MW for the L1. The allowed value range is [0,100].
kind	1..1	RemedialActionKind	(NC) inherited from: RemedialAction
penaltyFactor	0..1	Float	(NC) inherited from: RemedialAction
isCrossBorderRelevant	1..1	Boolean	(NC) inherited from: RemedialAction
isManual	0..1	Boolean	(NC) inherited from: RemedialAction
timeToImplement	0..1	Duration	(NC) inherited from: RemedialAction
impactThresholdMargin	0..1	PerCent	(NC) inherited from: RemedialAction
normalAvailable	1..1	Boolean	(NC) inherited from: RemedialAction
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

648 Table 9 shows all association ends of CountertradeRemedialAction with other classes.

649 **Table 9 – Association ends of RemedialActionProfile::CountertradeRemedialAction with**
650 **other classes**
651

mult from	name	mult to	type	description
0..*	BiddingZone	0..1	BiddingZone	(NC) inherited from: PowerRemedialAction
0..*	BiddingZoneBorder	0..1	BiddingZoneBorder	(NC) inherited from: PowerRemedialAction
0..*	AppointedToRegion	1..1	Region	(NC) inherited from: RemedialAction
0..*	RemedialActionSystemOperator	0..1	SystemOperator	(NC) inherited from: RemedialAction

652 3.10 (abstract) Contingency root class

653 An event threatening system reliability, consisting of one or more contingency elements.

654 3.11 (NC) ContingencyWithRemedialAction root class

655 Combination of a contingency and a remedial action. ContingencyWithRemedialAction shall not
656 be instantiated for preventive RemedialAction (RemedialAction.kind equals
657 RemedialActionKind.preventive).
658

Table 10 shows all attributes of ContingencyWithRemedialAction.

Table 10 – Attributes of RemedialActionProfile::ContingencyWithRemedialAction

name	mult	type	description
mRID	1..1	String	(NC) Master resource identifier issued by a model authority. The mRID is unique within an exchange context. Global uniqueness is easily achieved by using a UUID, as specified in RFC 4122, for the mRID. The use of UUID is strongly recommended. For CIMXML data files in RDF syntax conforming to IEC 61970-552, the mRID is mapped to rdf:ID or rdf:about attributes that identify CIM object elements.
combinationConstraintKind	1..1	ElementCombinationConstraintKind	(NC) Defines the combination constraint of the Contingency and Remedial Action. If included, this remedial action can only be applied for this contingency. Else if excluded, this remedial action should not be used for this contingency. Else if considered, this remedial action can be considered for this contingency.
normalEnabled	0..1	Boolean	(NC) If true, the contingency with remedial action is enabled, otherwise it is disabled under normal operating conditions.

Table 11 shows all association ends of ContingencyWithRemedialAction with other classes.

Table 11 – Association ends of RemedialActionProfile::ContingencyWithRemedialAction with other classes

mult from	name	mult to	type	description
0..*	Contingency	1..1	Contingency	(NC) The contingency that is associated with a remedial action, i.e. the contingency that is the cause for the creation of a remedial action and justifies it or would usually be resolved with a remedial action.
0..*	RemedialAction	1..1	RemedialAction	(NC) The remedial action defined for this contingency and remedial action combination.

3.12 (NC) ControlFunctionBlockAction

Inheritance path = [SetPointAction](#) : [GridStateAlteration](#) : [IdentifiedObject](#)

Action for setting the control function block target values.

Table 12 shows all attributes of ControlFunctionBlockAction.

Table 12 – Attributes of RemedialActionProfile::ControlFunctionBlockAction

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 13 shows all association ends of ControlFunctionBlockAction with other classes.

Table 13 – Association ends of RemedialActionProfile::ControlFunctionBlockAction with other classes

mult from	name	mult to	type	description
0..*	ControlFunctionBlock	1..1	ControlFunctionBlock	(NC) The control function block that is associated with a ControlFunctionBlockAction.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

3.13 (abstract) DCTerminal root class

An electrical connection point to generic DC conducting equipment.

3.14 (abstract) EnergyConsumer root class

Generic user of energy - a point of consumption on the power system model.

EnergyConsumer.pfixed, .qfixed, .pfixedPct and .qfixedPct have meaning only if there is no LoadResponseCharacteristic associated with EnergyConsumer or if LoadResponseCharacteristic.exponentModel is set to False.

3.15 (abstract) EnergySource root class

A generic equivalent for an energy supplier on a transmission or distribution voltage level.

3.16 (NC) EnergySourceModification

Inheritance path = [SetPointAction](#) : [GridStateAlteration](#) : [IdentifiedObject](#)

Energy source action.

Table 14 shows all attributes of EnergySourceModification.

Table 14 – Attributes of RemedialActionProfile::EnergySourceModification

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 15 shows all association ends of EnergySourceModification with other classes.

Table 15 – Association ends of RemedialActionProfile::EnergySourceModification with other classes

mult from	name	mult to	type	description
0..*	EnergySource	1..1	EnergySource	(NC) The EnergySource which is associated with an EnergySourceAction.

mult from	name	mult to	type	description
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

3.17 (abstract) EquivalentInjection root class

This class represents equivalent injections (generation or load). Voltage regulation is allowed only at the point of connection.

3.18 (NC) EquivalentInjectionAction

Inheritance path = [SetPointAction](#) : [GridStateAlteration](#) : [IdentifiedObject](#)

Equivalent injection action.

Table 16 shows all attributes of EquivalentInjectionAction.

Table 16 – Attributes of RemedialActionProfile::EquivalentInjectionAction

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 17 shows all association ends of EquivalentInjectionAction with other classes.

Table 17 – Association ends of RemedialActionProfile::EquivalentInjectionAction with other classes

mult from	name	mult to	type	description
0..*	EquivalentInjection	1..1	EquivalentInjection	(NC) The EquivalentInjection that is associated with an action.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

3.19 (abstract) ExternalNetworkInjection root class

This class represents the external network and it is used for IEC 60909 calculations.

3.20 (NC) ExternalNetworkInjectionAction

Inheritance path = [SetPointAction](#) : [GridStateAlteration](#) : [IdentifiedObject](#)

External network injection action.

Table 18 shows all attributes of ExternalNetworkInjectionAction.

Table 18 – Attributes of RemedialActionProfile::ExternalNetworkInjectionAction

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 19 shows all association ends of ExternalNetworkInjectionAction with other classes.

Table 19 – Association ends of RemedialActionProfile::ExternalNetworkInjectionAction with other classes

mult from	name	mult to	type	description
0..*	ExternalNetworkInjection	1..1	ExternalNetworkInjection	(NC) The ExternalNetworkInjection that is associated with an action.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

3.21 (NC) Gate

Inheritance path = [IdentifiedObject](#)

Logical gate that supports a logical operation based on the input.

Table 20 shows all attributes of Gate.

Table 20 – Attributes of RemedialActionProfile::Gate

name	mult	type	description
kind	1..1	GateLogicKind	(NC) The logical operation of the gate.
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

3.22 (abstract,NC) GateInputPin

Inheritance path = [FunctionInputVariable](#) : [IdentifiedObject](#)

Input pin for a logical gate. The condition described in the input pin gives a logical true or false.

The result from measurement and calculation are converted to a true or false.

Table 21 shows all attributes of GateInputPin.

Table 21 – Attributes of RemedialActionProfile::GateInputPin

name	mult	type	description
absoluteValue	1..1	Boolean	(NC) Indicates if the absolute value is used for comparison. If true, use the absolute value. If false, use the complex value (vector).

name	mult	type	description
duration	1..1	Duration	(NC) The time duration for which the condition is satisfied before acting. Default is 0 seconds.
logicKind	0..1	LogicalOperatorsKind	(NC) The logical operator kind used for comparison.
negate	1..1	Boolean	(NC) Invert/negate the result of the comparison.
thresholdPercentage	0..1	PerCent	(NC) The threshold percentage that should be used for compare with the percentage change between input value and threshold value. The allowed value range is [0,100].
thresholdValue	0..1	Float	(NC) The threshold value that should be used for compare with the input value.
isValuePreFault	0..1	Boolean	(NC) Indicates if the gate input pin value is referring to the value prior to a fault (e.g. simulated by a contingency or due to a SIPS activation in a N-x-y case). If it is true, it means that the value is referring to pre-fault. If it is false or not populated, then it is post-fault.
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 22 shows all association ends of GateInputPin with other classes.

Table 22 – Association ends of RemedialActionProfile::GateInputPin with other classes

mult from	name	mult to	type	description
1..*	Gate	1..1	Gate	(NC) The Gate that has this input.

3.23 (abstract,NC) GridStateAlteration

Inheritance path = [IdentifiedObject](#)

Grid state alteration is a change of values describing state (operating point) of one element in the grid model compared to the base case.

Table 23 shows all attributes of GridStateAlteration.

Table 23 – Attributes of RemedialActionProfile::GridStateAlteration

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) The default/normal value used when other active signal/values are missing.
maximumPerDay	0..1	Integer	(NC) Maximum number of alterations per day.
minimumActivation	0..1	Duration	(NC) Minimum time duration between activating the same grid state alteration.
timePerStage	0..1	Duration	(NC) Time to implement a stage of a grid state alteration. If a grid state alteration consists of multiple stages (e.g. A step on a power transformer), this duration comes in addition to the timeToImplement and need to be multiplied by the number of stages. A stage can also be defined as MW in the case of regulating production.
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject

name	mult	type	description
name	0..1	String	inherited from: IdentifiedObject

Table 24 shows all association ends of GridStateAlteration with other classes.

Table 24 – Association ends of RemedialActionProfile::GridStateAlteration with other classes

mult from	name	mult to	type	description
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) The grid state alteration remedial action associated with a given grid state alteration.
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) The collection that has a GridStateAlteration.
0..*	PropertyReference	1..1	PropertyReference	(NC) The property reference for this grid state alteration.

3.24 (NC) GridStateAlterationCollection

Inheritance path = [IdentifiedObject](#)

A collection of grid state alterations.

Table 25 shows all attributes of GridStateAlterationCollection.

Table 25 – Attributes of RemedialActionProfile::GridStateAlterationCollection

name	mult	type	description
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

3.25 (NC) GridStateAlterationRemedialAction

Inheritance path = [RemedialAction](#) : [IdentifiedObject](#)

Grid state alteration remedial action describes one or many grid state alterations applied to a grid model state or a particular scenario in order to resolve one or more identified constraints.

Table 26 shows all attributes of GridStateAlterationRemedialAction.

Table 26 – Attributes of RemedialActionProfile::GridStateAlterationRemedialAction

name	mult	type	description
isActivationRestricted	1..1	Boolean	(NC) If true, all GridStateAlterations associated with the GridStateAlterationRemedialAction need to be available in order to activate the RA. If false, it means that even if one or more GridStateAlterations is/are not available then the rest of the GridStateAlterationRemedialAction can be activated. Being available means it is not in a contingency or out of service. A switch that has an action of opening is still available if it is open. GridStateAlteration.normalEnabled or GridStateAlteration.enabled equal to false does not make the GridStateAlterationRemedialAction not available even if GridStateAlteration as part of the remedial action cannot be used in a GridStateAlterationRemedialAction.
kind	1..1	RemedialActionKind	(NC) inherited from: RemedialAction
penaltyFactor	0..1	Float	(NC) inherited from: RemedialAction

name	mult	type	description
isCrossBorderRelevant	1..1	Boolean	(NC) inherited from: RemedialAction
isManual	0..1	Boolean	(NC) inherited from: RemedialAction
timeToImplement	0..1	Duration	(NC) inherited from: RemedialAction
impactThresholdMargin	0..1	PerCent	(NC) inherited from: RemedialAction
normalAvailable	1..1	Boolean	(NC) inherited from: RemedialAction
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 27 shows all association ends of GridStateAlterationRemedialAction with other classes.

Table 27 – Association ends of RemedialActionProfile::GridStateAlterationRemedialAction with other classes

mult from	name	mult to	type	description
0..*	AppointedToRegion	1..1	Region	(NC) inherited from: RemedialAction
0..*	RemedialActionSystemOperator	0..1	SystemOperator	(NC) inherited from: RemedialAction

3.26 (abstract) IdentifiedObject root class

This is a root class to provide common identification for all classes needing identification and naming attributes.

Table 28 shows all attributes of IdentifiedObject.

Table 28 – Attributes of RemedialActionProfile::IdentifiedObject

name	mult	type	description
description	0..1	String	The description is a free human readable text describing or naming the object. It may be non unique and may not correlate to a naming hierarchy.
mRID	1..1	String	Master resource identifier issued by a model authority. The mRID is unique within an exchange context. Global uniqueness is easily achieved by using a UUID, as specified in RFC 4122, for the mRID. The use of UUID is strongly recommended. For CIMXML data files in RDF syntax conforming to IEC 61970-552, the mRID is mapped to rdf:ID or rdf:about attributes that identify CIM object elements.
name	0..1	String	The name is any free human readable and possibly non unique text naming the object.

3.27 (NC) IntertemporalPropertyRange

Inheritance path = [RangeConstraint](#) : [IdentifiedObject](#)

It represents the intertemporal range, which means that this is the maximum change of an attribute value between two time stamps or per time unit (e.g. hour). Both up and down directions are defined by the direction attribute, i.e. There are different schedules per direction. The class is not instantiated for PropertyReference which refers to Boolean type attributes.

For instance the following example illustrates the approach:

- A tap changer related grid state alteration having two intertemporal range schedules.

- For a particular point in time, the value from up schedule is 6 and the value from down schedule is 3.
 - Then, the GridStateIntensity for the same point in time cannot be more than plus 6 taps from the current, or more than minus 3 taps from the current.
- Table 29 shows all attributes of IntertemporalPropertyRange.

Table 29 – Attributes of RemedialActionProfile::IntertemporalPropertyRange

name	mult	type	description
direction	1..1	RelativeDirectionKind	(NC) inherited from: RangeConstraint
valueKind	1..1	ValueOffsetKind	(NC) inherited from: RangeConstraint
normalValue	0..1	Float	(NC) inherited from: RangeConstraint
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 30 shows all association ends of IntertemporalPropertyRange with other classes.

Table 30 – Association ends of RemedialActionProfile::IntertemporalPropertyRange with other classes

mult from	name	mult to	type	description
0..*	GridStateAlteration	1..1	GridStateAlteration	(NC) inherited from: RangeConstraint

3.28 (NC) LoadAction

Inheritance path = [SetPointAction](#) : [GridStateAlteration](#) : [IdentifiedObject](#)

Load action.

Table 31 shows all attributes of LoadAction.

Table 31 – Attributes of RemedialActionProfile::LoadAction

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 32 shows all association ends of LoadAction with other classes.

Table 32 – Association ends of RemedialActionProfile::LoadAction with other classes

mult from	name	mult to	type	description
0..*	EnergyConsumer	1..1	EnergyConsumer	(NC) The EnergyConsumer that is associated with a load action.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration

mult from	name	mult to	type	description
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlterationCollection
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlterationCollection

3.29 (abstract) Measurement root class

A Measurement represents any measured, calculated or non-measured non-calculated quantity. Any piece of equipment may contain Measurements, e.g. a substation may have temperature measurements and door open indications, a transformer may have oil temperature and tank pressure measurements, a bay may contain a number of power flow measurements and a Breaker may contain a switch status measurement.

The PSR - Measurement association is intended to capture this use of Measurement and is included in the naming hierarchy based on EquipmentContainer. The naming hierarchy typically has Measurements as leaves, e.g. Substation-VoltageLevel-Bay-Switch-Measurement.

Some Measurements represent quantities related to a particular sensor location in the network, e.g. a voltage transformer (VT) or potential transformer (PT) at a busbar or a current transformer (CT) at the bar between a breaker and an isolator. The sensing position is not captured in the PSR - Measurement association. Instead it is captured by the Measurement - Terminal association that is used to define the sensing location in the network topology. The location is defined by the connection of the Terminal to ConductingEquipment.

If both a Terminal and PSR are associated, and the PSR is of type ConductingEquipment, the associated Terminal should belong to that ConductingEquipment instance.

When the sensor location is needed both Measurement-PSR and Measurement-Terminal are used. The Measurement-Terminal association is never used alone.

3.30 (NC) MeasurementCalculator

Inheritance path = [IdentifiedObject](#)

Result of a calculation of one or more measurement.

Table 33 shows all attributes of MeasurementCalculator.

Table 33 – Attributes of RemedialActionProfile::MeasurementCalculator

name	mult	type	description
kind	1..1	CalculationKind	(NC) Calculation operation executed on the operands.
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

3.31 (NC) MeasurementCalculatorInput

Inheritance path = [IdentifiedObject](#)

Input to measurement calculation. It supports Analog, Discrete and Accumulator measurements.

Table 34 shows all attributes of MeasurementCalculatorInput.

Table 34 – Attributes of RemedialActionProfile::MeasurementCalculatorInput

name	mult	type	description
absoluteValue	1..1	Boolean	(NC) Indicates if the absolute value is used for comparison. If true, use the absolute value. If false, use the complex value (vector).

name	mult	type	description
order	1..1	Integer	(NC) Positive number that defines the order of the operand in the calculation. 0 means default in which case the order is not relevant.
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 35 shows all association ends of MeasurementCalculatorInput with other classes.

Table 35 – Association ends of RemedialActionProfile::MeasurementCalculatorInput with other classes

mult from	name	mult to	type	description
0..*	Measurement	1..1	Measurement	(NC) Measurement used as input to a calculation.
1..*	MeasurementCalculator	1..1	MeasurementCalculator	(NC) The measurement calculator using this calculator input.

3.32 (NC) PinContingency

Inheritance path = [GateInputPin](#) : [FunctionInputVariable](#) : [IdentifiedObject](#)

Input pin associated with a Contingency. It is used for comparison.

Table 36 shows all attributes of PinContingency.

Table 36 – Attributes of RemedialActionProfile::PinContingency

name	mult	type	description
absoluteValue	1..1	Boolean	(NC) inherited from: GateInputPin
duration	1..1	Duration	(NC) inherited from: GateInputPin
logicKind	0..1	LogicalOperatorsKind	(NC) inherited from: GateInputPin
negate	1..1	Boolean	(NC) inherited from: GateInputPin
thresholdPercentage	0..1	PerCent	(NC) inherited from: GateInputPin
thresholdValue	0..1	Float	(NC) inherited from: GateInputPin
isValuePreFault	0..1	Boolean	(NC) inherited from: GateInputPin
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 37 shows all association ends of PinContingency with other classes.

Table 37 – Association ends of RemedialActionProfile::PinContingency with other classes

mult from	name	mult to	type	description
0..*	Contingency	1..1	Contingency	(NC) The Contingency that is used in the input pin.
1..*	Gate	1..1	Gate	(NC) inherited from: GateInputPin

3.33 (NC) PinDCTerminal

Inheritance path = [GateInputPin](#) : [FunctionInputVariable](#) : [IdentifiedObject](#)

Input pin associated with a DCTerminal. It is used for comparison.

Table 38 shows all attributes of PinDCTerminal.

Table 38 – Attributes of RemedialActionProfile::PinDCTerminal

name	mult	type	description
kind	1..1	PinDCTerminalKind	(NC) The kind of quantity which is used as an input value.
absoluteValue	1..1	Boolean	(NC) inherited from: GateInputPin
duration	1..1	Duration	(NC) inherited from: GateInputPin
logicKind	0..1	LogicalOperatorsKind	(NC) inherited from: GateInputPin
negate	1..1	Boolean	(NC) inherited from: GateInputPin
thresholdPercentage	0..1	PerCent	(NC) inherited from: GateInputPin
thresholdValue	0..1	Float	(NC) inherited from: GateInputPin
isValuePreFault	0..1	Boolean	(NC) inherited from: GateInputPin
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 39 shows all association ends of PinDCTerminal with other classes.

Table 39 – Association ends of RemedialActionProfile::PinDCTerminal with other classes

mult from	name	mult to	type	description
0..*	DCTerminal	0..1	DCTerminal	(NC) The DC terminal that has this pin DC terminal.
1..*	Gate	1..1	Gate	(NC) inherited from: GateInputPin

3.34 (NC) PinEquipment

Inheritance path = [GateInputPin](#) : [FunctionInputVariable](#) : [IdentifiedObject](#)

Input pin associated with an Equipment. It is used for the comparison.

Table 40 shows all attributes of PinEquipment.

Table 40 – Attributes of RemedialActionProfile::PinEquipment

name	mult	type	description
absoluteValue	1..1	Boolean	(NC) inherited from: GateInputPin
duration	1..1	Duration	(NC) inherited from: GateInputPin
logicKind	0..1	LogicalOperatorsKind	(NC) inherited from: GateInputPin
negate	1..1	Boolean	(NC) inherited from: GateInputPin
thresholdPercentage	0..1	PerCent	(NC) inherited from: GateInputPin
thresholdValue	0..1	Float	(NC) inherited from: GateInputPin
isValuePreFault	0..1	Boolean	(NC) inherited from: GateInputPin
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject

name	mult	type	description
name	0..1	String	inherited from: IdentifiedObject

Table 41 shows all association ends of PinEquipment with other classes.

Table 41 – Association ends of RemedialActionProfile::PinEquipment with other classes

mult from	name	mult to	type	description
0..*	PropertyReference	1..1	PropertyReference	(NC) The property reference for this pin equipment.
0..*	Equipment	1..1	Equipment	(NC) The Equipment that is used in the input pin.
1..*	Gate	1..1	Gate	(NC) inherited from: GateInputPin

3.35 (NC) PinGate

Inheritance path = [GateInputPin](#) : [FunctionInputVariable](#) : [IdentifiedObject](#)

An output from one gate represents an input to another gate.

Table 42 shows all attributes of PinGate.

Table 42 – Attributes of RemedialActionProfile::PinGate

name	mult	type	description
absoluteValue	1..1	Boolean	(NC) inherited from: GateInputPin
duration	1..1	Duration	(NC) inherited from: GateInputPin
logicKind	0..1	LogicalOperatorsKind	(NC) inherited from: GateInputPin
negate	1..1	Boolean	(NC) inherited from: GateInputPin
thresholdPercentage	0..1	PerCent	(NC) inherited from: GateInputPin
thresholdValue	0..1	Float	(NC) inherited from: GateInputPin
isValuePreFault	0..1	Boolean	(NC) inherited from: GateInputPin
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 43 shows all association ends of PinGate with other classes.

Table 43 – Association ends of RemedialActionProfile::PinGate with other classes

mult from	name	mult to	type	description
0..*	GateOutput	1..1	Gate	(NC) The output of the gate.
1..*	Gate	1..1	Gate	(NC) inherited from: GateInputPin

3.36 (NC) PinMeasurement

Inheritance path = [GateInputPin](#) : [FunctionInputVariable](#) : [IdentifiedObject](#)

Input pin associated with a Measurement. It is used for comparison.

Table 44 shows all attributes of PinMeasurement.

876

Table 44 – Attributes of RemedialActionProfile::PinMeasurement

name	mult	type	description
absoluteValue	1..1	Boolean	(NC) inherited from: GateInputPin
duration	1..1	Duration	(NC) inherited from: GateInputPin
logicKind	0..1	LogicalOperatorsKind	(NC) inherited from: GateInputPin
negate	1..1	Boolean	(NC) inherited from: GateInputPin
thresholdPercentage	0..1	PerCent	(NC) inherited from: GateInputPin
thresholdValue	0..1	Float	(NC) inherited from: GateInputPin
isValuePreFault	0..1	Boolean	(NC) inherited from: GateInputPin
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

877

878

Table 45 shows all association ends of PinMeasurement with other classes.

879

Table 45 – Association ends of RemedialActionProfile::PinMeasurement with other classes

880

mult from	name	mult to	type	description
0..*	Measurement	0..1	Measurement	(NC) The Measurement that is used in the input pin.
0..*	MeasurementCalculator	0..1	MeasurementCalculator	(NC) The result of the calculation used as input to a gate.
1..*	Gate	1..1	Gate	(NC) inherited from: GateInputPin

881

3.37 (NC) PinPowerTransferCorridorInheritance path = [GateInputPin](#) : [FunctionInputVariable](#) : [IdentifiedObject](#)

Input pin associated with a PowerTransferCorridor. It is used for comparison.

Table 46 shows all attributes of PinPowerTransferCorridor.

886

Table 46 – Attributes of RemedialActionProfile::PinPowerTransferCorridor

name	mult	type	description
kind	1..1	PinPowerTransferCorridorKind	(NC) The kind of quantity which is used as an input value.
absoluteValue	1..1	Boolean	(NC) inherited from: GateInputPin
duration	1..1	Duration	(NC) inherited from: GateInputPin
logicKind	0..1	LogicalOperatorsKind	(NC) inherited from: GateInputPin
negate	1..1	Boolean	(NC) inherited from: GateInputPin
thresholdPercentage	0..1	PerCent	(NC) inherited from: GateInputPin
thresholdValue	0..1	Float	(NC) inherited from: GateInputPin
isValuePreFault	0..1	Boolean	(NC) inherited from: GateInputPin
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

887

888

Table 47 shows all association ends of PinPowerTransferCorridor with other classes.

Table 47 – Association ends of RemedialActionProfile::PinPowerTransferCorridor with other classes

mult from	name	mult to	type	description
0..*	PowerTransferCorridor	1..1	PowerTransferCorridor	(NC) The PowerTransferCorridor that is used in the input pin.
1..*	Gate	1..1	Gate	(NC) inherited from: GateInputPin

3.38 (NC) PinTerminal

Inheritance path = [GateInputPin](#) : [FunctionInputVariable](#) : [IdentifiedObject](#)

Input pin associated with a Terminal. It is used for comparison.

Table 48 shows all attributes of PinTerminal.

Table 48 – Attributes of RemedialActionProfile::PinTerminal

name	mult	type	description
kind	1..1	PinTerminalKind	(NC) The kind of quantity which is used as an input value.
absoluteValue	1..1	Boolean	(NC) inherited from: GateInputPin
duration	1..1	Duration	(NC) inherited from: GateInputPin
logicKind	0..1	LogicalOperatorsKind	(NC) inherited from: GateInputPin
negate	1..1	Boolean	(NC) inherited from: GateInputPin
thresholdPercentage	0..1	PerCent	(NC) inherited from: GateInputPin
thresholdValue	0..1	Float	(NC) inherited from: GateInputPin
isValuePreFault	0..1	Boolean	(NC) inherited from: GateInputPin
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 49 shows all association ends of PinTerminal with other classes.

Table 49 – Association ends of RemedialActionProfile::PinTerminal with other classes

mult from	name	mult to	type	description
0..*	Terminal	1..1	Terminal	(NC) The Terminal that is used in the input pin.
1..*	Gate	1..1	Gate	(NC) inherited from: GateInputPin

3.39 (NC) PinOperationalLimit

Inheritance path = [GateInputPin](#) : [FunctionInputVariable](#) : [IdentifiedObject](#)

Input pin associated with the limits of a Terminal. It is used for comparison.

Table 50 shows all attributes of PinOperationalLimit.

Table 50 – Attributes of RemedialActionProfile::PinOperationalLimit

name	mult	type	description
absoluteValue	1..1	Boolean	(NC) inherited from: GateInputPin
duration	1..1	Duration	(NC) inherited from: GateInputPin
logicKind	0..1	LogicalOperatorsKind	(NC) inherited from: GateInputPin

name	mult	type	description
negate	1..1	Boolean	(NC) inherited from: GateInputPin
thresholdPercentage	0..1	PerCent	(NC) inherited from: GateInputPin
thresholdValue	0..1	Float	(NC) inherited from: GateInputPin
isValuePreFault	0..1	Boolean	(NC) inherited from: GateInputPin
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 51 shows all association ends of PinOperationalLimit with other classes.

Table 51 – Association ends of RemedialActionProfile::PinOperationalLimit with other classes

mult from	name	mult to	type	description
0..*	OperationalLimit	1..1	OperationalLimit	(NC) The operational limit that is used in the input pin.
1..*	Gate	1..1	Gate	(NC) inherited from: GateInputPin

3.40 (abstract) PowerElectronicsConnection root class

A connection to the AC network for energy production or consumption that uses power electronics rather than rotating machines.

3.41 (NC) PowerElectronicsConnectionAction

Inheritance path = [SetPointAction](#) : [GridStateAlteration](#) : [IdentifiedObject](#)

Power electronics setpoint action.

Table 52 shows all attributes of PowerElectronicsConnectionAction.

Table 52 – Attributes of RemedialActionProfile::PowerElectronicsConnectionAction

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 53 shows all association ends of PowerElectronicsConnectionAction with other classes.

Table 53 – Association ends of RemedialActionProfile::PowerElectronicsConnectionAction with other classes

mult from	name	mult to	type	description
0..*	PowerElectronicsConnection	1..1	PowerElectronicsConnection	(NC) The PowerElectronicsConnection that is applied to an action.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration

mult from	name	mult to	type	description
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlterationCollection
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlterationCollection

3.42 (abstract) PowerSystemResource

Inheritance path = [IdentifiedObject](#)

A power system resource (PSR) can be an item of equipment such as a switch, an equipment container containing many individual items of equipment such as a substation, or an organisational entity such as sub-control area. Power system resources can have measurements associated.

Table 54 shows all attributes of PowerSystemResource.

Table 54 – Attributes of RemedialActionProfile::PowerSystemResource

name	mult	type	description
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

3.43 (abstract,NC) PowerTransferCorridor root class

A power transfer corridor is defined as a set of circuits (transmission lines or transformers) separating two portions of the power system, or a subset of circuits exposed to a substantial portion of the transmission exchange between two parts of the system.

3.44 (abstract,NC) PropertyReference root class

The reference to a class and one of its properties.

3.45 (NC) PTCActivePowerSupport root class

Defines the active power capability (support) of the scheme in relation to a PowerTransferCorridor.

Table 55 shows all attributes of PTCActivePowerSupport.

Table 55 – Attributes of RemedialActionProfile::PTCActivePowerSupport

name	mult	type	description
maximum	1..1	ActivePower	(NC) Maximum support that a System Integrity Protection Scheme (SIPS) can provide to a Power Transfer Corridor (PTC). This is normally limited by the maximum power system disconnect allowed.
normal	1..1	ActivePower	(NC) Normal support that a System Integrity Protection Scheme (SIPS) is expected to provide when enabled to a Power Transfer Corridor (PTC).

Table 56 shows all association ends of PTCActivePowerSupport with other classes.

Table 56 – Association ends of RemedialActionProfile::PTCActivePowerSupport with other classes

mult from	name	mult to	type	description
0..*	PowerTransferCorridor	1..1	PowerTransferCorridor	(NC) The PowerTransferCorridor that has a specific active power support.

mult from	name	mult to	type	description
0..*	RemedialActionScheme	1..1	RemedialActionScheme	(NC) The RemedialActionScheme which has active power support from the PowerTransferCorridor.

3.46 (NC) RedispatchRemedialAction

Inheritance path = [PowerRemedialAction](#) : [RemedialAction](#) : [IdentifiedObject](#)

Redispatch remedial action is a remedial action that through rearranging power schedules is eliminating breaches of constraints.

Table 57 shows all attributes of RedispatchRemedialAction.

Table 57 – Attributes of RemedialActionProfile::RedispatchRemedialAction

name	mult	type	description
kind	1..1	RemedialActionKind	(NC) inherited from: RemedialAction
penaltyFactor	0..1	Float	(NC) inherited from: RemedialAction
isCrossBorderRelevant	1..1	Boolean	(NC) inherited from: RemedialAction
isManual	0..1	Boolean	(NC) inherited from: RemedialAction
timeToImplement	0..1	Duration	(NC) inherited from: RemedialAction
impactThresholdMargin	0..1	PerCent	(NC) inherited from: RemedialAction
normalAvailable	1..1	Boolean	(NC) inherited from: RemedialAction
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 58 shows all association ends of RedispatchRemedialAction with other classes.

Table 58 – Association ends of RemedialActionProfile::RedispatchRemedialAction with other classes

mult from	name	mult to	type	description
0..*	BiddingZone	0..1	BiddingZone	(NC) inherited from: PowerRemedialAction
0..*	BiddingZoneBorder	0..1	BiddingZoneBorder	(NC) inherited from: PowerRemedialAction
0..*	AppointedToRegion	1..1	Region	(NC) inherited from: RemedialAction
0..*	RemedialActionSystemOperator	0..1	SystemOperator	(NC) inherited from: RemedialAction

3.47 (abstract,NC) Region root class

A region where the system operator belongs to.

3.48 (abstract) RegulatingControl root class

Specifies a set of equipment that works together to control a power system quantity such as voltage or flow.

Remote bus voltage control is possible by specifying the controlled terminal located at some place remote from the controlling equipment.

The specified terminal shall be associated with the connectivity node of the controlled point.

The most specific subtype of RegulatingControl shall be used in case such equipment participate in the control, e.g. TapChangerControl for tap changers.

For flow control, load sign convention is used, i.e. positive sign means flow out from a TopologicalNode (bus) into the conducting equipment.

The attribute minAllowedTargetValue and maxAllowedTargetValue are required in the following cases:

- For a power generating module operated in power factor control mode to specify maximum and minimum power factor values;

- Whenever it is necessary to have an off center target voltage for the tap changer regulator. For instance, due to long cables to off shore wind farms and the need to have a simpler setup at the off shore transformer platform, the voltage is controlled from the land at the connection point for the off shore wind farm. Since there usually is a voltage rise along the cable, there is typical and overvoltage of up 3-4 kV compared to the on shore station. Thus in normal operation the tap changer on the on shore station is operated with a target set point, which is in the lower parts of the dead band.

The attributes minAllowedTargetValue and maxAllowedTargetValue are not related to the attribute targetDeadband and thus they are not treated as an alternative of the targetDeadband. They are needed due to limitations in the local substation controller. The attribute targetDeadband is used to prevent the power flow from move the tap position in circles (hunting) that is to be used regardless of the attributes minAllowedTargetValue and maxAllowedTargetValue.

3.49 (NC) RegulatingControlAction

Inheritance path = [GridStateAlteration](#) : [IdentifiedObject](#)

Control action means the set point change of a regulating control power system resource in the grid model compared to the base case.

Table 59 shows all attributes of RegulatingControlAction.

Table 59 – Attributes of RemedialActionProfile::RegulatingControlAction

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 60 shows all association ends of RegulatingControlAction with other classes.

Table 60 – Association ends of RemedialActionProfile::RegulatingControlAction with other classes

mult from	name	mult to	type	description
0..*	RegulatingControl	1..1	RegulatingControl	(NC) The regulating control which has an action.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

3.50 (abstract,NC) RemedialAction

Inheritance path = [IdentifiedObject](#)

1002 Remedial action describes one or more actions that can be performed on a given power system
 1003 model situation to eliminate one or more identified breaches of constraints. The remedial action
 1004 can be costly, and have a cost characteristic, or non costly.
 1005 Table 61 shows all attributes of RemedialAction.

1006 **Table 61 – Attributes of RemedialActionProfile::RemedialAction**

name	mult	type	description
kind	1..1	RemedialActionKind	(NC) The kind of the remedial action. If curative remedial action, it is required to have an association with ContingencyWithRemedialAction. If preventive remedial action, RemedialAction class shall not have association with ContingencyWithRemedialAction.
penaltyFactor	0..1	Float	(NC) Defines the relative penalty for a given remedial action. This is a positive number greater than zero and default is one, meaning the remedial action does not have negative nor positive effect on the quality of the solution. A remedial action that provide changes in the transmission loss can have negative (Between zero and one) or positive effect (Bigger than one) given by $1 / (1 - \text{Incremental Transmission Loss})$. In a similar way remedial action using generating units or compensation units can have negative or positive effect. Typical value would be between 0.8 and 1.1.
isCrossBorderRelevant	1..1	Boolean	(NC) Indicates if the remedial action is cross border relevant. True, means that the remedial action is cross border relevant.
isManual	0..1	Boolean	(NC) Indicates if the remedial action is manually executed which involves one or many actions performed by human. A SIPS remedial action cannot be manual. True, means that the remedial action is manual. False, means that the remedial action is automatically executed without human communication.
timeToImplement	0..1	Duration	(NC) Time to implement a remedial action.
impactThresholdMargin	0..1	PerCent	(NC) Impact threshold margin for the use of the remedial action. Meaning that the remedial action should not be used if it cannot resolve violation with more than the given impact threshold margin. The allowed value range is [0,100].
normalAvailable	1..1	Boolean	(NC) Identifies if the remedial action can be used in further processes under normal condition. True means available, False means unavailable. For instance, remedial action optimisation and coordination.
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

1007
 1008 Table 62 shows all association ends of RemedialAction with other classes.

Table 62 – Association ends of RemedialActionProfile::RemedialAction with other classes

mult from	name	mult to	type	description
0..*	AppointedToRegion	1..1	Region	(NC) The region in which the remedial action is appointed.
0..*	RemedialActionSystemOperator	0..1	SystemOperator	(NC) System operator operating remedial actions.

3.51 (NC) RemedialActionScheme

Inheritance path = [PowerSystemResource](#) : [IdentifiedObject](#)

Remedial Action Scheme (RAS), Special Protection Schemes (SPS), System Protection Schemes (SPS) or System Integrity Protection Schemes (SIPS).

A Remedial Action Scheme consists of one or more stages that can trigger and execute a protection action.

Table 63 shows all attributes of RemedialActionScheme.

Table 63 – Attributes of RemedialActionProfile::RemedialActionScheme

name	mult	type	description
kind	1..1	RemedialActionSchemeKind	(NC) Kind of Remedial Action Scheme.
normalArmed	1..1	Boolean	(NC) Defines the normal arming status of the remedial action scheme.
normallyInService	0..1	Boolean	(NC) Indicates if the remedial action scheme and related assets are normally active and can participate in system operation under normal operating conditions. If true, the scheme is normally active. If false, the scheme is treated as if it is not in the model.
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 64 shows all association ends of RemedialActionScheme with other classes.

Table 64 – Association ends of RemedialActionProfile::RemedialActionScheme with other classes

mult from	name	mult to	type	description
0..*	GateArmed	0..1	Gate	(NC) Gate that through a gate logic and input pin defines arming of a Remedial Action Scheme.
1..1	SchemeRemedialAction	0..1	SchemeRemedialAction	(NC) Scheme remedial action that belongs to the remedial action scheme.

3.52 (abstract) RotatingMachine root class

A rotating machine which may be used as a generator or motor.

3.53 (NC) RotatingMachineAction

Inheritance path = [SetPointAction](#) : [GridStateAlteration](#) : [IdentifiedObject](#)

Rotating machine action.

Table 65 shows all attributes of RotatingMachineAction.

1031 **Table 65 – Attributes of RemedialActionProfile::RotatingMachineAction**

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

1032
1033 Table 66 shows all association ends of RotatingMachineAction with other classes.

1034 **Table 66 – Association ends of RemedialActionProfile::RotatingMachineAction with**
1035 **other classes**

mult from	name	mult to	type	description
0..*	RotatingMachine	1..1	RotatingMachine	(NC) The rotating machine that has an action.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

1036
1037 **3.54 (NC) SchemeRemedialAction**

1038 Inheritance path = [RemedialAction](#) : [IdentifiedObject](#)

1039 Scheme remedial action is remedial action that involves a scheme that can include conditional
1040 logic and stages of grid alteration. The primary remedial action is the arming of these schemes,
1041 that will then perform curative remedial action when the condition is met. System Integrity
1042 Protection Scheme (SIPS) and Special Protection Scheme (SPS) are example of this.
1043 Table 67 shows all attributes of SchemeRemedialAction.

1044 **Table 67 – Attributes of RemedialActionProfile::SchemeRemedialAction**

name	mult	type	description
kind	1..1	RemedialActionKind	(NC) inherited from: RemedialAction
penaltyFactor	0..1	Float	(NC) inherited from: RemedialAction
isCrossBorderRelevant	1..1	Boolean	(NC) inherited from: RemedialAction
isManual	0..1	Boolean	(NC) inherited from: RemedialAction
timeToImplement	0..1	Duration	(NC) inherited from: RemedialAction
impactThresholdMargin	0..1	PerCent	(NC) inherited from: RemedialAction
normalAvailable	1..1	Boolean	(NC) inherited from: RemedialAction
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

1045
1046 Table 68 shows all association ends of SchemeRemedialAction with other classes.

Table 68 – Association ends of RemedialActionProfile::SchemeRemedialAction with other classes

mult from	name	mult to	type	description
0..*	AppointedToRegion	1..1	Region	(NC) inherited from: RemedialAction
0..*	RemedialActionSystemOperator	0..1	SystemOperator	(NC) inherited from: RemedialAction

3.55 (abstract,NC) SetPointAction

Inheritance path = [GridStateAlteration](#) : [IdentifiedObject](#)

Setpoint action.

Table 69 shows all attributes of SetPointAction.

Table 69 – Attributes of RemedialActionProfile::SetPointAction

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 70 shows all association ends of SetPointAction with other classes.

Table 70 – Association ends of RemedialActionProfile::SetPointAction with other classes

mult from	name	mult to	type	description
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

3.56 (abstract) ShuntCompensator root class

A shunt capacitor or reactor or switchable bank of shunt capacitors or reactors. A section of a shunt compensator is an individual capacitor or reactor. A negative value for bPerSection indicates that the compensator is a reactor. ShuntCompensator is a single terminal device. Ground is implied.

3.57 (NC) ShuntCompensatorModification

Inheritance path = [SetPointAction](#) : [GridStateAlteration](#) : [IdentifiedObject](#)

Shunt compensator action.

Table 71 shows all attributes of ShuntCompensatorModification.

Table 71 – Attributes of RemedialActionProfile::ShuntCompensatorModification

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration

name	mult	type	description
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 72 shows all association ends of ShuntCompensatorModification with other classes.

Table 72 – Association ends of RemedialActionProfile::ShuntCompensatorModification with other classes

mult from	name	mult to	type	description
0..*	ShuntCompensator	1..1	ShuntCompensator	(NC) The ShuntCompensator that is associated with an action.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

3.58 (NC) Stage

Inheritance path = [IdentifiedObject](#)

Stage of a remedial action scheme.

Table 73 shows all attributes of Stage.

Table 73 – Attributes of RemedialActionProfile::Stage

name	mult	type	description
priority	1..1	Integer	(NC) The priority of the stage. 0 = do not care (default) 1 = highest priority. 2 is less than 1 and so on. A stage with higher priority needs be activated before a lower stage can be activated.
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 74 shows all association ends of Stage with other classes.

Table 74 – Association ends of RemedialActionProfile::Stage with other classes

mult from	name	mult to	type	description
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) The GridStateAlterationCollection which belongs to the Stage.
1..*	RemedialActionScheme	1..1	RemedialActionScheme	(NC) The remedial action scheme that has a stage.

3.59 (NC) StageTrigger

Inheritance path = [PowerSystemResource](#) : [IdentifiedObject](#)

1086 Stage that is triggered either by TriggerCondition or by gate condition within a stage.
1087 Table 75 shows all attributes of StageTrigger.

1088 **Table 75 – Attributes of RemedialActionProfile::StageTrigger**

name	mult	type	description
normalArmed	1..1	Boolean	(NC) The default/normal value used when other active signal/values are missing.
priority	1..1	Integer	(NC) Priority of trigger. 0 = don t care (default) 1 = highest priority. 2 is less than 1 and so on. A trigger with the highest priority will trigger first.
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

1089
1090 Table 76 shows all association ends of StageTrigger with other classes.

1091 **Table 76 – Association ends of RemedialActionProfile::StageTrigger with other classes**

mult from	name	mult to	type	description
0..*	GateComCondition	0..1	Gate	(NC) The gate that is the input pin which defines a communication condition.
0..*	GateArmed	0..1	Gate	(NC) The gate that is the input pin which defines arming of the StageTrigger.
0..*	GateTrigger	0..1	Gate	(NC) The gate that is the input pin which triggers the protective reactions.
1..*	Stage	1..1	Stage	(NC) The stage that has this stage trigger.

1092 1093 3.60 (NC) StaticPropertyRange

1094 Inheritance path = [RangeConstraint](#) : [IdentifiedObject](#)

1095 Defines the static range, which means that this is the minimum and/or maximum of an attribute
1096 value. The value provided by the schedule replaces the value of the attribute to which the
1097 schedule refers to.

1098 In case that the PropertyReference refers to Boolean type attributes, RangeConstraint.direction
1099 shall be none or upAndDown and the RangeConstraint.valueKind shall be absolute. If the
1100 direction is none then optimization of the attribute referenced by the PropertyReference is not
1101 possible if the current status is already as the value in the range. Otherwise if the direction is
1102 upAndDown, the optimization can change from true to false or vice versa independently of the
1103 initial value in the operational scenario.

1104 For instance for a tap changer related grid state alteration for a particular point in time, if the
1105 range of TapChanger.step is to be restricted, the value of the schedule will represent that new
1106 TapChanger.step range.

1107 Table 77 shows all attributes of StaticPropertyRange.

1108 **Table 77 – Attributes of RemedialActionProfile::StaticPropertyRange**

name	mult	type	description
direction	1..1	RelativeDirectionKind	(NC) inherited from: RangeConstraint
valueKind	1..1	ValueOffsetKind	(NC) inherited from: RangeConstraint
normalValue	0..1	Float	(NC) inherited from: RangeConstraint
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject

name	mult	type	description
name	0..1	String	inherited from: IdentifiedObject

Table 78 shows all association ends of StaticPropertyRange with other classes.

Table 78 – Association ends of RemedialActionProfile::StaticPropertyRange with other classes

mult from	name	mult to	type	description
0..*	PropertyReference	1..1	PropertyReference	Property reference for this static property range.
0..*	GridStateAlteration	1..1	GridStateAlteration	(NC) inherited from: RangeConstraint

3.61 (abstract) StaticVarCompensator root class

A facility for providing variable and controllable shunt reactive power. The SVC typically consists of a stepdown transformer, filter, thyristor-controlled reactor, and thyristor-switched capacitor arms.

The SVC may operate in fixed MVar output mode or in voltage control mode. When in voltage control mode, the output of the SVC will be proportional to the deviation of voltage at the controlled bus from the voltage setpoint. The SVC characteristic slope defines the proportion. If the voltage at the controlled bus is equal to the voltage setpoint, the SVC MVar output is zero.

3.62 (NC) StaticVarCompensatorAction

Inheritance path = [SetPointAction](#) : [GridStateAlteration](#) : [IdentifiedObject](#)

Static Var compensator action.

Table 79 shows all attributes of StaticVarCompensatorAction.

Table 79 – Attributes of RemedialActionProfile::StaticVarCompensatorAction

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 80 shows all association ends of StaticVarCompensatorAction with other classes.

Table 80 – Association ends of RemedialActionProfile::StaticVarCompensatorAction with other classes

mult from	name	mult to	type	description
0..*	StaticVarCompensator	1..1	StaticVarCompensator	(NC) The StaticVarCompensator which is associated with an action.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

1131

1132 **3.63 (Description) Substation root class**

1133 A collection of equipment for purposes other than generation or utilization, through which
1134 electric energy in bulk is passed for the purposes of switching or modifying its characteristics.

1135 **3.64 (abstract) Switch root class**

1136 A generic device designed to close, or open, or both, one or more electric circuits. All switches
1137 are two terminal devices including grounding switches. The ACDCTerminal.connected at the
1138 two sides of the switch shall not be considered for assessing switch connectivity, i.e. only
1139 Switch.open, .normalOpen and .locked are relevant.

1140 **3.65 (abstract,NC) SystemOperator root class**

1141 System operator.

1142 **3.66 (abstract) TapChanger root class**

1143 Mechanism for changing transformer winding tap positions.

1144 **3.67 (NC) TapPositionAction**

1145 Inheritance path = [GridStateAlteration](#) : [IdentifiedObject](#)

1146 Tap position action represents a change of a tap changer position in the grid model compared
1147 to the base case.

1148 Table 81 shows all attributes of TapPositionAction.

1149 **Table 81 – Attributes of RemedialActionProfile::TapPositionAction**

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

1150

1151 Table 82 shows all association ends of TapPositionAction with other classes.

1152 **Table 82 – Association ends of RemedialActionProfile::TapPositionAction with other**
1153 **classes**

mult from	name	mult to	type	description
0..*	TapChanger	1..1	TapChanger	(NC) The tap changer that has a tap position action associated.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

1154

1155 **3.68 (abstract) Terminal root class**

1156 An AC electrical connection point to a piece of conducting equipment. Terminals are connected
1157 at physical connection points called connectivity nodes.

1158 **3.69 (NC) TopologyAction**1159 Inheritance path = [GridStateAlteration](#) : [IdentifiedObject](#)1160 Topology action means the connection or disconnection of a switch in the grid model compared
1161 to the base case.

1162 Table 83 shows all attributes of TopologyAction.

1163 **Table 83 – Attributes of RemedialActionProfile::TopologyAction**

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

1164

1165 Table 84 shows all association ends of TopologyAction with other classes.

1166 **Table 84 – Association ends of RemedialActionProfile::TopologyAction with other**
1167 **classes**

mult from	name	mult to	type	description
0..*	Switch	1..1	Switch	(NC) The switch that has a topology action associated.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

1168

1169 **3.70 (deprecated,NC) TriggerCondition**1170 Inheritance path = [IdentifiedObject](#)

1171 The condition that triggers a remedial action scheme.

1172 Table 85 shows all attributes of TriggerCondition.

1173 **Table 85 – Attributes of RemedialActionProfile::TriggerCondition**

name	mult	type	description
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

1174

1175 Table 86 shows all association ends of TriggerCondition with other classes.

1176 **Table 86 – Association ends of RemedialActionProfile::TriggerCondition with other**
1177 **classes**

mult from	name	mult to	type	description
0..*	GateTrigger	1..1	Gate	(deprecated,NC) The gate that is the condition for the trigger.

mult from	name	mult to	type	description
0..*	RemedialActionScheme	1..1	RemedialActionScheme	(deprecated,NC) The remedial action scheme that has the trigger condition.

1178

1179 **3.71 (NC) CalculationKind enumeration**

1180 Kind of calculation operation that can be done to Measurement.

1181 Table 87 shows all literals of CalculationKind.

1182 **Table 87 – Literals of RemedialActionProfile::CalculationKind**

literal	value	description
summation		Summation operation on the input values (operands).
multiplication		Multiplication operation on the input values (operands).
division		Division operation on the input values (operands).
squareRoot		Square root operator - only one input value (operands).

1183

1184 **3.72 Currency enumeration**

1185 Monetary currencies. ISO 4217 standard including 3-character currency code.

1186 Table 88 shows all literals of Currency.

1187 **Table 88 – Literals of RemedialActionProfile::Currency**

literal	value	description
AED	784	United Arab Emirates dirham.
AFN	971	Afghan afghani.
ALL	008	Albanian lek.
AMD	051	Armenian dram.
ANG	532	Netherlands Antillean guilder.
AOA	973	Angolan kwanza.
ARS	032	Argentine peso.
AUD	036	Australian dollar.
AWG	533	Aruban florin.
AZN	944	Azerbaijani manat.
BAM	977	Bosnia and Herzegovina convertible mark.
BBD	052	Barbados dollar.
BDT	050	Bangladeshi taka.
BGN	975	Bulgarian lev.
BHD	048	Bahraini dinar.
BIF	108	Burundian franc.
BMD	060	Bermudian dollar (customarily known as Bermuda dollar).
BND	096	Brunei dollar.
BOB	068	Boliviano.

literal	value	description
BOV	984	Bolivian Mvdol (funds code).
BRL	986	Brazilian real.
BSD	044	Bahamian dollar.
BTN	064	Bhutanese ngultrum.
BWP	072	Botswana pula.
BYR	974	Belarusian ruble.
BZD	084	Belize dollar.
CAD	124	Canadian dollar.
CDF	976	Congolese franc.
CHF	756	Swiss franc.
CLF	990	Unidad de Fomento (funds code), Chile.
CLP	152	Chilean peso.
CNY	156	Chinese yuan.
COP	170	Colombian peso.
COU	970	Unidad de Valor Real.
CRC	188	Costa Rican colon.
CUC	931	Cuban convertible peso.
CUP	192	Cuban peso.
CVE	132	Cape Verde escudo.
CZK	203	Czech koruna.
DJF	262	Djiboutian franc.
DKK	208	Danish krone.
DOP	214	Dominican peso.
DZD	012	Algerian dinar.
EEK	233	Estonian kroon.
EGP	818	Egyptian pound.
ERN	232	Eritrean nakfa.
ETB	230	Ethiopian birr.
EUR	978	Euro.
FJD	242	Fiji dollar.
FKP	238	Falkland Islands pound.
GBP	826	Pound sterling.
GEL	981	Georgian lari.
GHS	936	Ghanaian cedi.
GIP	929	Gibraltar pound.
GMD	270	Gambian dalasi.
GNF	324	Guinean franc.
GTQ	320	Guatemalan quetzal.
GYD	328	Guyanese dollar.
HKD	344	Hong Kong dollar.
HNL	340	Honduran lempira.

literal	value	description
HRK	191	Croatian kuna.
HTG	332	Haitian gourde.
HUF	348	Hungarian forint.
IDR	360	Indonesian rupiah.
ILS	376	Israeli new sheqel.
INR	356	Indian rupee.
IQD	368	Iraqi dinar.
IRR	364	Iranian rial.
ISK	352	Icelandic króna.
JMD	388	Jamaican dollar.
JOD	400	Jordanian dinar.
JPY	392	Japanese yen.
KES	404	Kenyan shilling.
KGS	417	Kyrgyzstani som.
KHR	116	Cambodian riel.
KMF	174	Comoro franc.
KPW	408	North Korean won.
KRW	410	South Korean won.
KWD	414	Kuwaiti dinar.
KYD	136	Cayman Islands dollar.
KZT	398	Kazakhstani tenge.
LAK	418	Lao kip.
LBP	422	Lebanese pound.
LKR	144	Sri Lanka rupee.
LRD	430	Liberian dollar.
LSL	426	Lesotho loti.
LTL	440	Lithuanian litas.
LVL	428	Latvian lats.
LYD	434	Libyan dinar.
MAD	504	Moroccan dirham.
MDL	498	Moldovan leu.
MGA	969	Malagasy ariary.
MKD	807	Macedonian denar.
MMK	104	Myanma kyat.
MNT	496	Mongolian tugrik.
MOP	446	Macanese pataca.
MRO	478	Mauritanian ouguiya.
MUR	480	Mauritian rupee.
MVR	462	Maldivian rufiyaa.
MWK	454	Malawian kwacha.
MXN	484	Mexican peso.

literal	value	description
MYR	458	Malaysian ringgit.
MZN	943	Mozambican metical.
NAD	516	Namibian dollar.
NGN	566	Nigerian naira.
NIO	558	Cordoba oro.
NOK	578	Norwegian krone.
NPR	524	Nepalese rupee.
NZD	554	New Zealand dollar.
OMR	512	Omani rial.
PAB	590	Panamanian balboa.
PEN	604	Peruvian nuevo sol.
PGK	598	Papua New Guinean kina.
PHP	608	Philippine peso.
PKR	586	Pakistani rupee.
PLN	985	Polish zloty.
PYG	600	Paraguayan guaraní.
QAR	634	Qatari rial.
RON	946	Romanian new leu.
RSD	941	Serbian dinar.
RUB	643	Russian rouble.
RWF	646	Rwandan franc.
SAR	682	Saudi riyal.
SBD	090	Solomon Islands dollar.
SCR	690	Seychelles rupee.
SDG	938	Sudanese pound.
SEK	752	Swedish krona/kronor.
SGD	702	Singapore dollar.
SHP	654	Saint Helena pound.
SLL	694	Sierra Leonean leone.
SOS	706	Somali shilling.
SRD	968	Surinamese dollar.
STD	678	São Tomé and Príncipe dobra.
SYP	760	Syrian pound.
SZL	748	Lilangeni.
THB	764	Thai baht.
TJS	972	Tajikistani somoni.
TMT	934	Turkmenistani manat.
TND	788	Tunisian dinar.
TOP	776	Tongan pa'anga.
TRY	949	Turkish lira.
TTD	780	Trinidad and Tobago dollar.

literal	value	description
TWD	901	New Taiwan dollar.
TZS	834	Tanzanian shilling.
UAH	980	Ukrainian hryvnia.
UGX	800	Ugandan shilling.
USD	840	United States dollar.
UYU	858	Uruguayan peso.
UZS	860	Uzbekistan som.
VEF	937	Venezuelan bolívar fuerte.
VND	704	Vietnamese Dong.
VUV	548	Vanuatu vatu.
WST	882	Samoan tala.
XAF	950	CFA franc BEAC.
XCD	951	East Caribbean dollar.
XOF	952	CFA Franc BCEAO.
XPF	953	CFP franc.
YER	886	Yemeni rial.
ZAR	710	South African rand.
ZMK	894	Zambian kwacha.
ZWL	932	Zimbabwe dollar.

1188

1189 **3.73 (NC) GateLogicKind enumeration**

1190 Define the different logical operations.

1191 Table 89 shows all literals of GateLogicKind.

1192

Table 89 – Literals of RemedialActionProfile::GateLogicKind

literal	value	description
and		A logical AND operation. True when all inputs are true.
or		A logical OR operation. True when one or more inputs are true.
nor		A logical NOR operation. False when one or more inputs are true.
nand		A logical NAND operation. False when all inputs are true.
not		A logical NOT operation. Only one input and true input will give false out and false in will give true out. An inverter.
xnor		A logical XNOR operation. The function is the inverse of the exclusive OR (XOR) gate. All input false or true will give true. Otherwise false.
xor		A logical XOR operation. All input false or true will give false. Otherwise true.

1193

1194 **3.74 (NC) LogicalOperatorsKind enumeration**

1195 Kinds of logical operators for comparison.

1196 Table 90 shows all literals of LogicalOperatorsKind.

1197 **Table 90 – Literals of RemedialActionProfile::LogicalOperatorsKind**

literal	value	description
notEqual		Not equal (unlike) comparison operation.
equals		Equals (like) comparison operation.
lessThanOrEquals		Less than or equals comparison operation.
lessThan		Less than comparison operation.
greaterThanOrEquals		Greater than or equals comparison operation.
greaterThan		Greater than comparison operation.

1198

1199 **3.75 (NC) PinDCTerminalKind enumeration**

1200 The kind of quantities that can serve as an input value for the DCTerminal pin.

1201 Table 91 shows all literals of PinDCTerminalKind.

1202 **Table 91 – Literals of RemedialActionProfile::PinDCTerminalKind**

literal	value	description
voltage		Direct current voltage in the DCTerminal.
current		Direct current in the DCTerminal.

1203

1204 **3.76 (NC) PinPowerTransferCorridorKind enumeration**

1205 The kind of quantities that can serve as an input value for the PowerTransferCorridor pin.

1206 Table 92 shows all literals of PinPowerTransferCorridorKind.

1207 **Table 92 – Literals of RemedialActionProfile::PinPowerTransferCorridorKind**

literal	value	description
activePower		Active power in the branch group.
reactivePower		Reactive power in the branch group.

1208

1209 **3.77 (NC) RelativeDirectionKind enumeration**

1210 Kind of direction for the changes.

1211 Table 93 shows all literals of RelativeDirectionKind.

1212 **Table 93 – Literals of RemedialActionProfile::RelativeDirectionKind**

literal	value	description
up		Up signifies that the changes are increasing from the current status.
down		Down signifies that the changes are decreasing from the current status.
upAndDown		Up and down signifies that both up and down values are equal.
none		There is no direction on the changes.

1213

1214 **3.78 (NC) RemedialActionKind enumeration**

1215 The different kinds for a remedial action.

1216 Table 94 shows all literals of RemedialActionKind.

1217 **Table 94 – Literals of RemedialActionProfile::RemedialActionKind**

literal	value	description
curative		Curative remedial action means a remedial action that is the result of an operational planning process and is activated straight subsequent to the occurrence of the respective contingency for compliance with the (N-1) criterion, taking into account transitory admissible overloads and their accepted duration.
preventive		Preventive remedial action means a remedial action that is the result of an operational planning process and needs to be activated prior to the investigated timeframe for compliance with the (N-1) criterion.

1218

1219 **3.79 (NC) RemedialActionSchemeKind enumeration**

1220 Classification of Remedial Action Scheme.

1221 Table 95 shows all literals of RemedialActionSchemeKind.

1222 **Table 95 – Literals of RemedialActionProfile::RemedialActionSchemeKind**

literal	value	description
sips		System Integrity Protection Scheme (SIPS). The triggering conditions are met through field measurements.
rasp		Remedial Action Schema Plan (RASP). The triggering conditions are met through calculation or manual intervention.

1223

1224 **3.80 UnitSymbol enumeration**

1225 The derived units defined for usage in the CIM. In some cases, the derived unit is equal to an
 1226 SI unit. Whenever possible, the standard derived symbol is used instead of the formula for the
 1227 derived unit. For example, the unit symbol Farad is defined as "F" instead of "CPerV". In cases
 1228 where a standard symbol does not exist for a derived unit, the formula for the unit is used as
 1229 the unit symbol. For example, density does not have a standard symbol and so it is represented
 1230 as "kgPerm3". With the exception of the "kg", which is an SI unit, the unit symbols do not contain
 1231 multipliers and therefore represent the base derived unit to which a multiplier can be applied as
 1232 a whole.

1233 Every unit symbol is treated as an unparseable text as if it were a single-letter symbol. The
 1234 meaning of each unit symbol is defined by the accompanying descriptive text and not by the
 1235 text contents of the unit symbol.

1236 To allow the widest possible range of serializations without requiring special character handling,
 1237 several substitutions are made which deviate from the format described in IEC 80000-1. The
 1238 division symbol "/" is replaced by the letters "Per". Exponents are written in plain text after the
 1239 unit as "m3" instead of being formatted as "m" with a superscript of 3 or introducing a symbol
 1240 as in "m^3". The degree symbol "°" is replaced with the letters "deg". Any clarification of the
 1241 meaning for a substitution is included in the description for the unit symbol.

1242 Non-SI units are included in list of unit symbols to allow sources of data to be correctly labelled
 1243 with their non-SI units (for example, a GPS sensor that is reporting numbers that represent feet
 1244 instead of meters). This allows software to use the unit symbol information correctly convert
 1245 and scale the raw data of those sources into SI-based units.

1246 The integer values are used for harmonization with IEC 61850.

1247 Table 96 shows all literals of UnitSymbol.

1248

Table 96 – Literals of RemedialActionProfile::UnitSymbol

literal	value	description
none	0	Dimension less quantity, e.g. count, per unit, etc.
s	4	Time in seconds.
W	38	Real power in watts (J/s). Electrical power may have real and reactive components. The real portion of electrical power (I^2R or $VI\cos(\phi)$), is expressed in Watts. See also apparent power and reactive power.
VA _r	63	Reactive power in volt amperes reactive. The "reactive" or "imaginary" component of electrical power ($VI\sin(\phi)$). (See also real power and apparent power). Note: Different meter designs use different methods to arrive at their results. Some meters may compute reactive power as an arithmetic value, while others compute the value vectorially. The data consumer should determine the method in use and the suitability of the measurement for the intended purpose.

1249

1250 3.81 UnitMultiplier enumeration

1251 The unit multipliers defined for the CIM. When applied to unit symbols, the unit symbol is
 1252 treated as a derived unit. Regardless of the contents of the unit symbol text, the unit symbol
 1253 shall be treated as if it were a single-character unit symbol. Unit symbols should not contain
 1254 multipliers, and it should be left to the multiplier to define the multiple for an entire data type.
 1255 For example, if a unit symbol is "m2Pers" and the multiplier is "k", then the value is $k(m^{**2}/s)$,
 1256 and the multiplier applies to the entire final value, not to any individual part of the value. This
 1257 can be conceptualized by substituting a derived unit symbol for the unit type. If one imagines
 1258 that the symbol "P" represents the derived unit "m2Pers", then applying the multiplier "k" can
 1259 be conceptualized simply as "kP".

1260 For example, the SI unit for mass is "kg" and not "g". If the unit symbol is defined as "kg", then
 1261 the multiplier is applied to "kg" as a whole and does not replace the "k" in front of the "g". In
 1262 this case, the multiplier of "m" would be used with the unit symbol of "kg" to represent one gram.
 1263 As a text string, this violates the instructions in IEC 80000-1. However, because the unit symbol
 1264 in CIM is treated as a derived unit instead of as an SI unit, it makes more sense to conceptualize
 1265 the "kg" as if it were replaced by one of the proposed replacements for the SI mass symbol. If
 1266 one imagines that the "kg" were replaced by a symbol "P", then it is easier to conceptualize the
 1267 multiplier "m" as creating the proper unit "mP", and not the forbidden unit "mkg".

1268 Table 97 shows all literals of UnitMultiplier.

1269

Table 97 – Literals of RemedialActionProfile::UnitMultiplier

literal	value	description
none	0	No multiplier or equivalently multiply by 1.
M	6	Mega 10^{**6} .

1270

1271 3.82 (NC) ValueOffsetKind enumeration

1272 The kind of the value offset.

1273 Table 98 shows all literals of ValueOffsetKind.

1274 **Table 98 – Literals of RemedialActionProfile::ValueOffsetKind**

literal	value	description
absolute		Value of the range constraint is replacing the attribute value referenced by the PropertyReference in a determined operational scenario.
incremental		Value of the range constraint is incrementing the attribute value referenced by the PropertyReference in a determined operational scenario.
incrementalPercentage		Value of the range constraint is incrementing in percentage the attribute value referenced by the PropertyReference in a determined operational scenario.

1275

1276 **3.83 ActivePower datatype**1277 Product of RMS value of the voltage and the RMS value of the in-phase component of the
1278 current.

1279 Table 99 shows all attributes of ActivePower.

1280 **Table 99 – Attributes of RemedialActionProfile::ActivePower**

name	mult	type	description
multiplier	0..1	UnitMultiplier	(const=M)
unit	0..1	UnitSymbol	(const=W)
value	0..1	Float	

1281

1282 **3.84 PerCent datatype**

1283 Percentage on a defined base. For example, specify as 100 to indicate at the defined base.

1284 Table 100 shows all attributes of PerCent.

1285 **Table 100 – Attributes of RemedialActionProfile::PerCent**

name	mult	type	description
value	0..1	Float	Normally 0 to 100 on a defined base.
unit	0..1	UnitSymbol	(const=none)
multiplier	0..1	UnitMultiplier	(const=none)

1286

1287 **3.85 ReactivePower datatype**1288 Product of RMS value of the voltage and the RMS value of the quadrature component of the
1289 current.

1290 Table 101 shows all attributes of ReactivePower.

1291 **Table 101 – Attributes of RemedialActionProfile::ReactivePower**

name	mult	type	description
value	0..1	Float	
unit	0..1	UnitSymbol	(const=VAr)
multiplier	0..1	UnitMultiplier	(const=M)

1292

1293 **3.86 Seconds datatype**

1294 Time, in seconds.

1295 Table 102 shows all attributes of Seconds.

1296 **Table 102 – Attributes of RemedialActionProfile::Seconds**

name	mult	type	description
value	0..1	Float	Time, in seconds
unit	0..1	UnitSymbol	(const=s)
multiplier	0..1	UnitMultiplier	(const=none)

1297

1298 **3.87 Boolean primitive**

1299 A type with the value space "true" and "false".

1300 **3.88 Duration primitive**

1301 Duration as "PnYnMnDTnHnMnS" which conforms to ISO 8601, where nY expresses a number
 1302 of years, nM a number of months, nD a number of days. The letter T separates the date
 1303 expression from the time expression and, after it, nH identifies a number of hours, nM a number
 1304 of minutes and nS a number of seconds. The number of seconds could be expressed as a
 1305 decimal number, but all other numbers are integers.

1306 **3.89 Float primitive**

1307 A floating point number. The range is unspecified and not limited.

1308 **3.90 Integer primitive**

1309 An integer number. The range is unspecified and not limited.

1310 **3.91 String primitive**

1311 A string consisting of a sequence of characters. The character encoding is UTF-8. The string
 1312 length is unspecified and unlimited.

1313 **3.92 (NC) BiddingZoneAction**1314 Inheritance path = [SetPointAction](#) : [GridStateAlteration](#) : [IdentifiedObject](#)

1315 Bidding zone set point action.

1316 Table 103 shows all attributes of BiddingZoneAction.

1317 **Table 103 – Attributes of RemedialActionProfile::BiddingZoneAction**

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

1318

1319 Table 104 shows all association ends of BiddingZoneAction with other classes.

Table 104 – Association ends of RemedialActionProfile::BiddingZoneAction with other classes

mult from	name	mult to	type	description
0..*	BiddingZone	1..1	BiddingZone	(NC) The bidding zone that has this bidding zone action.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

3.93 (abstract,NC) ControlFunctionBlock root class

Control function block is a function block that contains an algorithm for controlling the equipment.

3.94 (abstract,NC) RangeConstraint

Inheritance path = [IdentifiedObject](#)

Defines the range constraint.

Table 105 shows all attributes of RangeConstraint.

Table 105 – Attributes of RemedialActionProfile::RangeConstraint

name	mult	type	description
direction	1..1	RelativeDirectionKind	(NC) Defines the direction of the attribute value referenced by the PropertyReference.
valueKind	1..1	ValueOffsetKind	(NC) Kind of value offset for the range that applies to the attribute referenced by the PropertyReference.
normalValue	0..1	Float	(NC) The normal (initial) value. The meaning of the value is defined by the attribute referenced by the PropertyReference. The attribute value can be integer, float or boolean. In case of boolean 1 equals true and 0 equals false. If the valueKind is incremental or incrementalPercentage, then the attribute value shall be positive (greater than zero). If the valueKind is absolute, it does not constraint attribute value to be positive. If the valueKind is incrementalPercentage, then the attribute value "100" shall represent 100%. However, the attribute value can be greater than 100 but not negative.
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 106 shows all association ends of RangeConstraint with other classes.

Table 106 – Association ends of RemedialActionProfile::RangeConstraint with other classes

mult from	name	mult to	type	description
0..*	GridStateAlteration	1..1	GridStateAlteration	(NC) The grid state alteration which has static range.

3.95 (NC) PinEquipmentTripping

Inheritance path = [GateInputPin](#) : [FunctionInputVariable](#) : [IdentifiedObject](#)

Input pin associated with an Equipment. It is used to determine if the equipment is tripped between two consecutive stages, i.e. the equipment is in service at pre-fault stage and it is out of service at post-fault stage.

Table 107 shows all attributes of PinEquipmentTripping.

Table 107 – Attributes of RemedialActionProfile::PinEquipmentTripping

name	mult	type	description
absoluteValue	1..1	Boolean	(NC) inherited from: GateInputPin
duration	1..1	Duration	(NC) inherited from: GateInputPin
logicKind	0..1	LogicalOperatorsKind	(NC) inherited from: GateInputPin
negate	1..1	Boolean	(NC) inherited from: GateInputPin
thresholdPercentage	0..1	PerCent	(NC) inherited from: GateInputPin
thresholdValue	0..1	Float	(NC) inherited from: GateInputPin
isValuePreFault	0..1	Boolean	(NC) inherited from: GateInputPin
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 108 shows all association ends of PinEquipmentTripping with other classes.

Table 108 – Association ends of RemedialActionProfile::PinEquipmentTripping with other classes

mult from	name	mult to	type	description
0..*	Equipment	1..1	Equipment	(NC) Equipment that is tripped.
1..*	Gate	1..1	Gate	(NC) inherited from: GateInputPin

3.96 (abstract) Equipment root class

The parts of a power system that are physical devices, electronic or mechanical.

3.97 (NC) RemedialActionDependency root class

Remedial action dependency is making two remedial actions depending on each other. Multiple dependency is done by multiple instances of this class. The dependency can arrive by having one of the following examples.

- The dependent remedial action is controlled by different system operator (Modeling Authority) (e.g. SIPS that goes across control area).

- The dependent remedial action is representing two or more remedial action that represent the same grid state alteration but with different modeling resolution (e.g. detail direct current model versus a simplified model).

- The remedial action can be combined with other remedial action without the need to create multiple remedial action with the same grid alteration for enabling dependency.

Table 109 shows all attributes of RemedialActionDependency.

1362 **Table 109 – Attributes of RemedialActionProfile::RemedialActionDependency**

name	mult	type	description
kind	1..1	RemedialActionDependencyKind	(NC) Type of dependency between two remedial actions.
mRID	1..1	String	(NC) Master resource identifier issued by a model authority. The mRID is unique within an exchange context. Global uniqueness is easily achieved by using a UUID, as specified in RFC 4122, for the mRID. The use of UUID is strongly recommended. For CIMXML data files in RDF syntax conforming to IEC 61970-552, the mRID is mapped to rdf:ID or rdf:about attributes that identify CIM object elements.
normalEnabled	0..1	Boolean	(NC) If true, the remedial action dependency within a group shall be considered under normal operating conditions.

1363 Table 110 shows all association ends of RemedialActionDependency with other classes.

1364 **Table 110 – Association ends of RemedialActionProfile::RemedialActionDependency with other classes**

mult from	name	mult to	type	description
0..*	RemedialAction	0..1	RemedialAction	(NC) Remedial action which has dependent remedial actions.
0..*	DependingRemedialActionGroup	0..1	RemedialActionGroup	(NC) Remedial action group which the remedial action is depending on.

1367 3.98 (abstract,NC) FunctionInputVariable

1368 Inheritance path = [IdentifiedObject](#)

1369 Functional input variable defines the domain of the function.

1370 Table 111 shows all attributes of FunctionInputVariable.

1371 **Table 111 – Attributes of RemedialActionProfile::FunctionInputVariable**

name	mult	type	description
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

1373 3.99 (NC) RemedialActionDependencyKind enumeration

1374 Kind of dependency between remedial actions.

1375 Table 112 shows all literals of RemedialActionDependencyKind.

1376 **Table 112 – Literals of RemedialActionProfile::RemedialActionDependencyKind**

literal	value	description
exclusive		Remedial actions are exclusive depending on each other. e.g. Only one of the remedial actions can be selected at the same time.
inclusive		Remedial actions are inclusive depending on each other. e.g. Both remedial actions need to be picked if one of them is needed.

literal	value	description
restrictive		Remedial actions are restrictive depending on each other. The need to include or to exclude might depend on the model. e.g. In the case of simplified DC model and detailed DC model. In the case where the simplified remedial action is used but not the remedial action for the detail model and opposite for the DC model.
none		Remedial actions are not depending on each other. However, the two remedial actions should be evaluated together.
balanced		This applies only to a set of power remedial actions (including HVDC which play role to balance power) and means that the remedial action needs to be balanced between the area (directly or indirectly to the bidding zone) that it is applied to.

1378

1379 **3.100 (abstract,NC) PowerRemedialAction**1380 Inheritance path = [RemedialAction](#) : [IdentifiedObject](#)

1381 Energy remedial action describes actions to rearrange power schedules.

1382 Table 113 shows all attributes of PowerRemedialAction.

1383 **Table 113 – Attributes of RemedialActionProfile::PowerRemedialAction**

name	mult	type	description
kind	1..1	RemedialActionKind	(NC) inherited from: RemedialAction
penaltyFactor	0..1	Float	(NC) inherited from: RemedialAction
isCrossBorderRelevant	1..1	Boolean	(NC) inherited from: RemedialAction
isManual	0..1	Boolean	(NC) inherited from: RemedialAction
timeToImplement	0..1	Duration	(NC) inherited from: RemedialAction
impactThresholdMargin	0..1	PerCent	(NC) inherited from: RemedialAction
normalAvailable	1..1	Boolean	(NC) inherited from: RemedialAction
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

1384

1385 Table 114 shows all association ends of PowerRemedialAction with other classes.

1386 **Table 114 – Association ends of RemedialActionProfile::PowerRemedialAction with other classes**

mult from	name	mult to	type	description
0..*	BiddingZone	0..1	BiddingZone	(NC) The Bidding Zone where the power remedial action is done.
0..*	BiddingZoneBorder	0..1	BiddingZoneBorder	(NC) Bidding zone border where the power remedial action is done.
0..*	AppointedToRegion	1..1	Region	(NC) inherited from: RemedialAction
0..*	RemedialActionSystemOperator	0..1	SystemOperator	(NC) inherited from: RemedialAction

1388

3.101 (NC) EquipmentControllerAction

Inheritance path = [SetPointAction](#) : [GridStateAlteration](#) : [IdentifiedObject](#)

Action for setting the equipment controller action.

Table 115 shows all attributes of EquipmentControllerAction.

Table 115 – Attributes of RemedialActionProfile::EquipmentControllerAction

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 116 shows all association ends of EquipmentControllerAction with other classes.

Table 116 – Association ends of RemedialActionProfile::EquipmentControllerAction with other classes

mult from	name	mult to	type	description
0..*	EquipmentController	1..1	EquipmentController	(NC) Equipment controller that has associated equipment controller actions.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

3.102 (abstract,NC) EquipmentController root class

Equipment controller is an automation function that can control one or multiple equipment function to achieve all the targets inside the given tolerance.

3.103 (NC) RemedialActionGroup

Inheritance path = [IdentifiedObject](#)

Grouping of remedial actions that can be operated together.

Table 117 shows all attributes of RemedialActionGroup.

Table 117 – Attributes of RemedialActionProfile::RemedialActionGroup

name	mult	type	description
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

3.104 (NC) ElementCombinationConstraintKind enumeration

Kind of constraint for an element combination.

Table 118 shows all literals of ElementCombinationConstraintKind.

1411 **Table 118 – Literals of RemedialActionProfile::ElementCombinationConstraintKind**

literal	value	description
included		Element combination is included.
excluded		Element combination is excluded.
considered		Element combination can be considered.

1412

1413 **3.105 (NC,Description) PowerShiftKeyStrategy**1414 Inheritance path = [IdentifiedObject](#)

1415 Strategy of the power shift key.

1416 Table 119 shows all attributes of PowerShiftKeyStrategy.

1417 **Table 119 – Attributes of RemedialActionProfile::PowerShiftKeyStrategy**

name	mult	type	description
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

1418

1419 Table 120 shows all association ends of PowerShiftKeyStrategy with other classes.

1420 **Table 120 – Association ends of RemedialActionProfile::PowerShiftKeyStrategy with**
1421 **other classes**

mult from	name	mult to	type	description
0..*	PowerRemedialAction	1..1	PowerRemedialAction	(NC) Power remedial action which has power shift key strategy.

1422

1423 **3.106 (abstract) OperationalLimit root class**

1424 A value and normal value associated with a specific kind of limit.

1425 The sub class value and normalValue attributes vary inversely to the associated
1426 OperationalLimitType.acceptableDuration (acceptableDuration for short).

1427 If a particular piece of equipment has multiple operational limits of the same kind (apparent
1428 power, current, etc.), the limit with the greatest acceptableDuration shall have the smallest limit
1429 value and the limit with the smallest acceptableDuration shall have the largest limit value. Note:
1430 A large current can only be allowed to flow through a piece of equipment for a short duration
1431 without causing damage, but a lesser current can be allowed to flow for a longer duration.

1432 **3.107 (abstract,NC) BiddingZoneBorder root class**

1433 A unidirectional interface between two adjacent Bidding Zones, representing the available
1434 transmission capacity from the Bidding Zone defined by the association FromBiddingZone
1435 (source) to the Bidding Zone defined by the association ToBiddingZone (sink). It defines the
1436 capacity constraints and market-based allocation mechanisms for cross-zonal electricity
1437 exchange in the specified direction, considering potential transmission congestion and market
1438 price differences.

1439 **3.108 (NC) PinTerminalKind enumeration**

1440 The kind of quantities that can serve as an input value for the pin.

1441 Table 121 shows all literals of PinTerminalKind.

1442 **Table 121 – Literals of RemedialActionProfile::PinTerminalKind**

literal	value	description
activePower		Active power on the Terminal.
apparentPower		Apparent power on the Terminal.
voltageMagnitude		Voltage magnitude on the Terminal.
voltageAngle		Voltage angle on the Terminal.
current		Current on the Terminal.
reactivePower		Reactive power on the Terminal.

1443

1444 **3.109 (NC) InServiceAction**1445 Inheritance path = [GridStateAlteration](#) : [IdentifiedObject](#)

1446 In service action represents a change of the status of the equipment in the grid model compared to the base case.

1447 Table 122 shows all attributes of InServiceAction.

1448 **Table 122 – Attributes of RemedialActionProfile::InServiceAction**

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) inherited from: GridStateAlteration
maximumPerDay	0..1	Integer	(NC) inherited from: GridStateAlteration
minimumActivation	0..1	Duration	(NC) inherited from: GridStateAlteration
timePerStage	0..1	Duration	(NC) inherited from: GridStateAlteration
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

1450

1451 Table 123 shows all association ends of InServiceAction with other classes.

1452 **Table 123 – Association ends of RemedialActionProfile::InServiceAction with other**
1453 **classes**

mult from	name	mult to	type	description
0..*	Equipment	1..1	Equipment	(NC) Equipment that has a in service action associated.
1..*	GridStateAlterationRemedialAction	0..1	GridStateAlterationRemedialAction	(NC) inherited from: GridStateAlteration
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

1454

1455

1456

1457 **Annex A (informative): Sample data**

1458 **A.1 General**

1459 This Annex is designed to illustrate the profile by using fragments of sample data. It is not meant
1460 to be a complete set of examples covering all possibilities of using the profile. Defining a
1461 complete set of test data is considered a separate activity to be performed for the purpose of
1462 setting up interoperability testing and conformity related to this profile.

1463 **A.2 Sample instance data**

1464 Test data files are available in the CIM EG SharePoint.

1465

1466