European Network of
Transmission System Operators
for Electricity

REMEDIAL ACTION PROFILE SPECIFICATION

2026-09-10

ICTC APPROVED
VERSION 2.5.0

Copyright notice:

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.

33

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.
2.5.0	2026-08-05		For CIM WG review.
2.5.0	2026-09-10		ICTC approval.

CONTENTS

List of figures

List of tables

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 precising 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-20v03_NC25v25_MS10v12_DES10v01.qea.

This edition is based on the IEC 61970 UML version ‘’, dated ‘’.

- Title:
- Keyword:
- Description:
- Version IRI:
- Version info:
- Prior version:
- Conforms to:
- Identifier:

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}”

where

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

[1] [SOURCE: CACM art.2.13]

[2] [SOURCE: 2019 Inter-RSC report]

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

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

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

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

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

2.3 Profile constraints

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

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

- C:452:ALL:NA:datatypes

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

- R:452:ALL:NA:exchange

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

- R:452:ALL:NA:exchange1

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

- R:452:ALL:NA:exchange2

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

- R:452:ALL:NA:exchange3

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

- R:452:ALL:NA:exchange4

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

- R:452:ALL:NA:cardinality

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

- R:452:ALL:NA:associations

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

- R:452:ALL:IdentifiedObject.name:rule

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

- R:452:ALL:IdentifiedObject.description:rule

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

- R:452:ALL:NA:uniqueIdentifier

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

- R:452:ALL:NA:unitMultiplier

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

- C:452:ALL:IdentifiedObject.name:stringLength

The string IdentifiedObject.name has a maximum of 128 characters.

- C:452:ALL:IdentifiedObject.description:stringLength

The string IdentifiedObject.description is maximum 256 characters.

- C:452:ALL:NA:float

An attribute that is defined as float (e.g. has a type Float or a type which is a Datatype with .value attribute of type Float) shall support ISO/IEC 60559:2020 for floating-point arithmetic using single precision floating point. A single precision float supports 7 significant digits where the significant digits are described as an integer, or a decimal number with 6 decimal digits. Two float values are equal when the significant with 7 digits are identical, e.g. 1234567 is equal 1.234567E6 and so are 1.2345678 and 1.234567E0.

- R:NC:ALL:Region:reference

The reference to the Region is normally a reference to the capacity calculation region, which is identified by “Y” EIC code of the capacity calculation region.

- R:NC:ALL:SystemOperator:reference

The reference to the System Operator is normally identified by “X” EIC code of TSO.

- R:NC:ALL:NA:serialization

The profiles are defined in the EnterpriseArchitect application and have multiple artifacts that describe them. The main artifacts are:

1) the EAP file (EnterpriseArchitect project file),
2) the profiles’ specification document and
3) the application profiles (RDFS and SHACL).

Due to the complexity of the profiles, there are various cross profile associations that, from profiling and profile maintenance point of view, it is not practical to include the complete inheritance structure in all profiles. If this is done the documentation provided for all profiles would also include duplicated information on the description of classes defined in other profiles. The following cases are often observed in profiles:

  - Case 1: An association end refers to an abstract class
  - Case 2: An abstract class (stereotyped with “Description”) has an association (direction to another class)
  - Case 3: An abstract class (not stereotyped with “Description”) has an association (direction to another class)
  - Case 4: An abstract class has attributes and subclasses are not in the profile

In all cases, the datasets shall only include the subtypes of the abstract classes with the related properties (i.e. association or attributes) defined in the profile. The information is taken from either canonical model or the profiles where complete (expected) inheritance structure for the related abstract class is described. SHACL based constraints include constraints only for the concrete classes that are subtypes of the abstract class in the profile, and this can be used to inform which are the concrete classes expected in a dataset that conforms to this profile.

It should be taken into account that this approach deviates from MVAL5 (IEC 61970-600-1:2021), which creates multiple inheritance at serialization. For instance, with this more explicit exchange the serialization of the association between abstract class Equipment and abstract class Circuit for a PowerTransformer will be serialized as follows:

  - for association

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

504 </cim:PowerTransformer>

505 ○ for attribute

506 <cim:ACLineSegment rdf:about="_04f681aa-6999-4fb3-9775-acaa5eb7ceff">

507 <cim:Equipment.inService>true</cim:Equipment.inService>

508 </cim:ACLineSegment>

509 The usage of rdf:ID or rdf:about depends on the stereotype of the class. rdf:about is
510 used if the class has the stereotype “Description”.

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

512 <cim:Equipment rdf:about="_c328f787-bc17-47ad-a59f-6ba7133340d0">

513 <nc:Equipment.Circut rdf:resource="#_9ced16ac-d076-4ef9-a241-a998a579e77b"/>

514 </cim:Equipment>

515 • C:NC:RA:GridStateAlteration:associations

516 A GridStateAlteration shall have either association to
517 GridStateAlterationRemedialAction or GridStateAlterationCollection.

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

519 The value of the association end StaticPropertyRange.PropertyReference shall be one
520 of the values published in the PropertyReference collection.

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

522 The value of the association end GridStateAlteration.PropertyReference shall be one of
523 the values published in the PropertyReference collection.

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

525 The value of the association end PinEquipment.PropertyReference shall be one of the
526 values published in the PropertyReference collection.

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

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

530

531

532 ## 2.4 Metadata

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

539 The header vocabulary contains all attributes defined in IEC 61970-552. This is done only for
540 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.

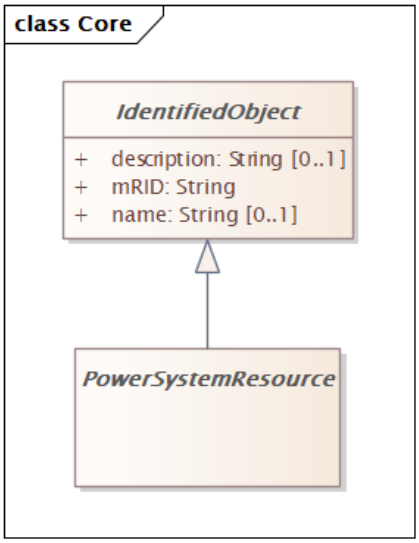
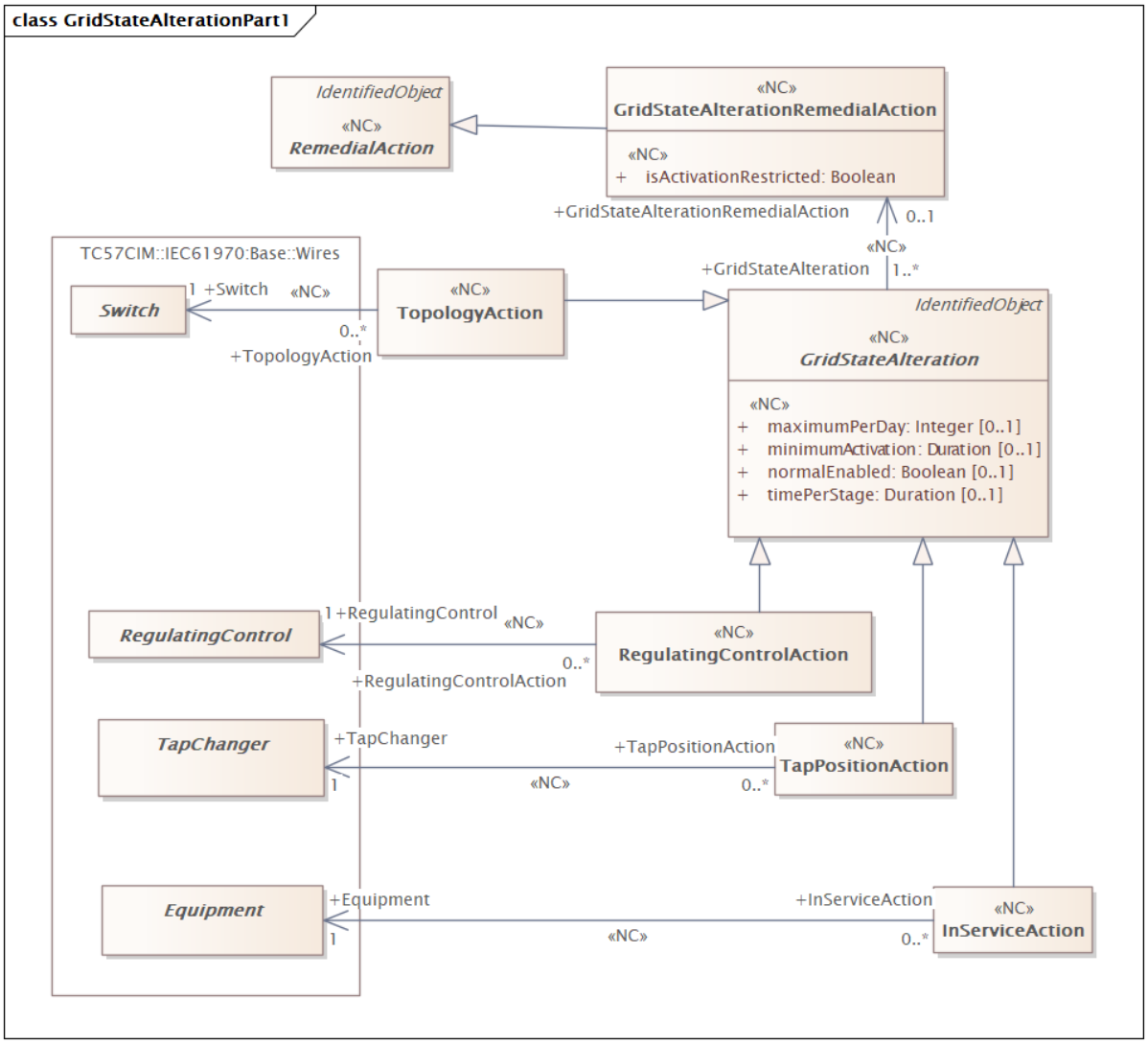


Figure 1 – Class diagram RemedialActionProfile::Core

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



565

566 **Figure 2 – Class diagram RemedialActionProfile::GridStateAlterationPart1**

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

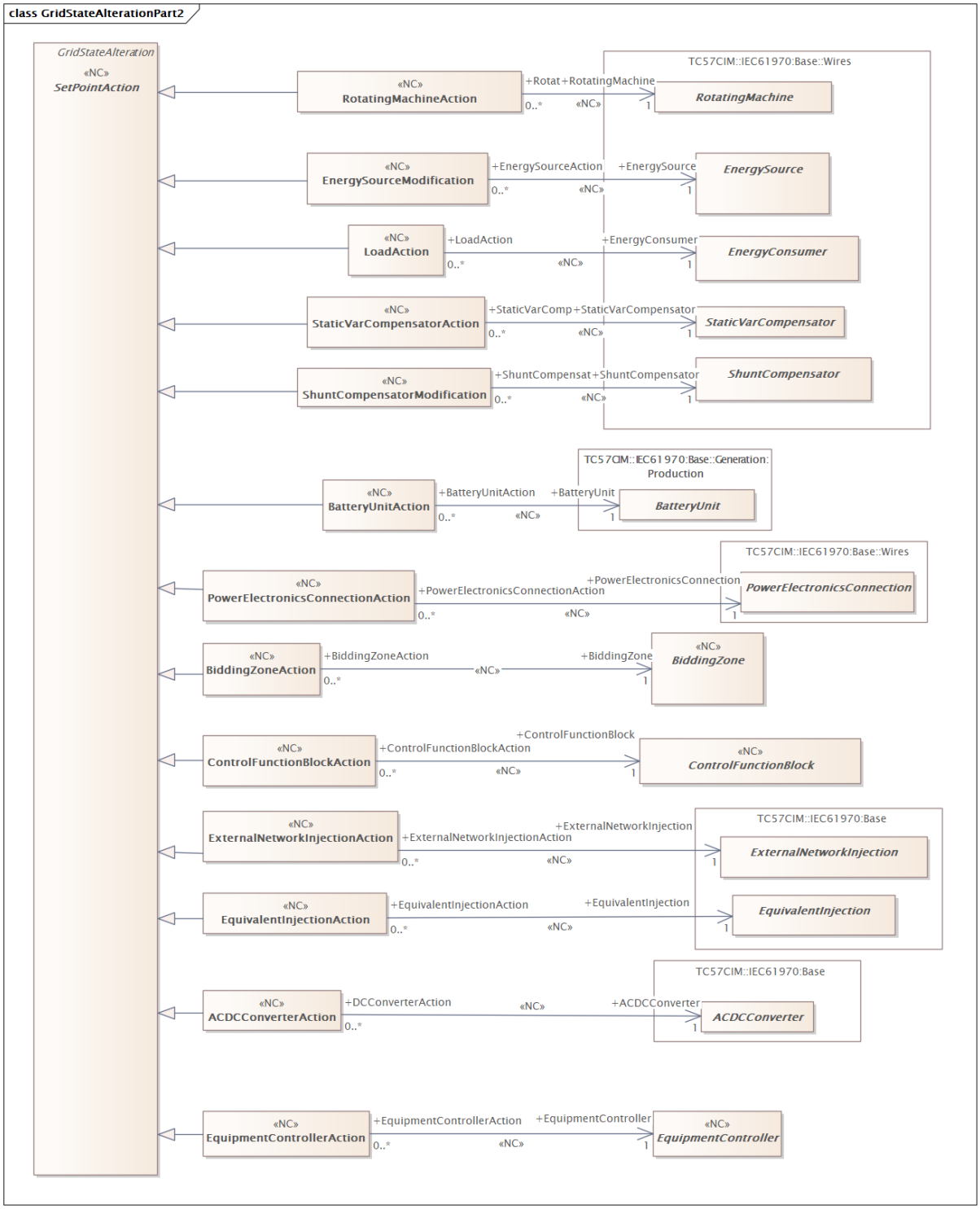


Figure 3 – Class diagram RemedialActionProfile::GridStateAlterationPart2

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

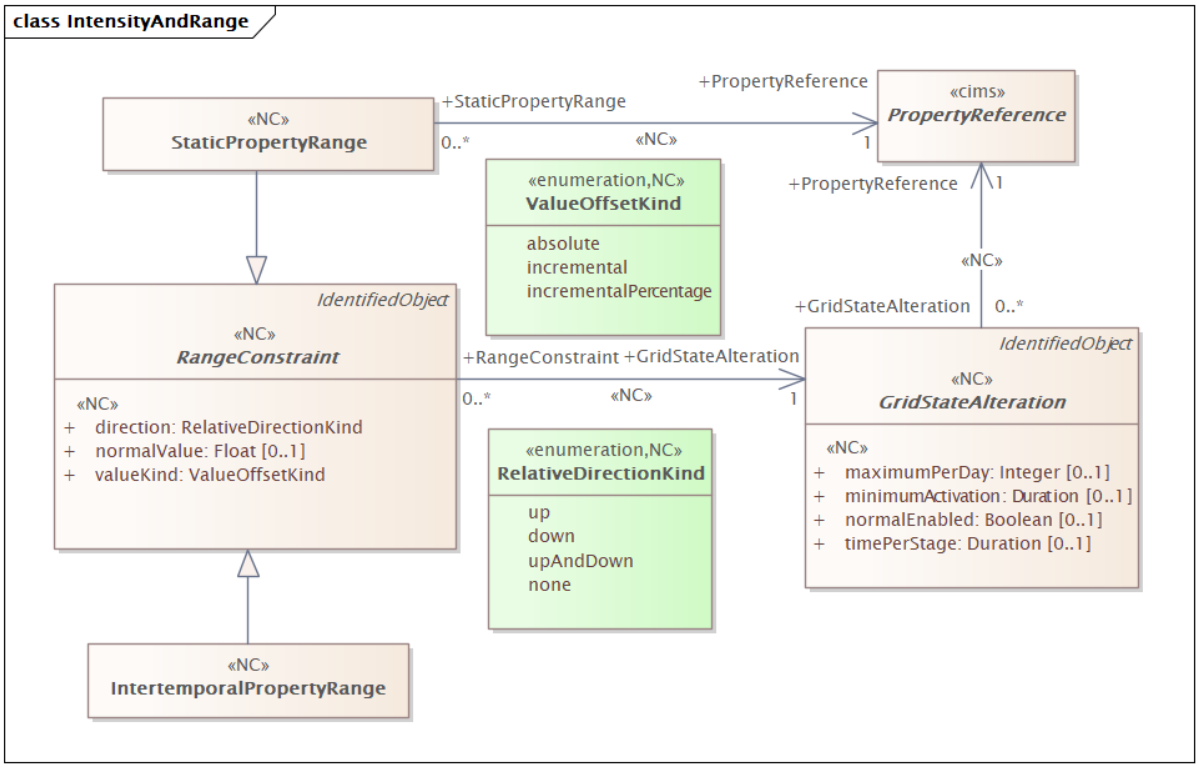


Figure 4 – Class diagram RemedialActionProfile::IntensityAndRange

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

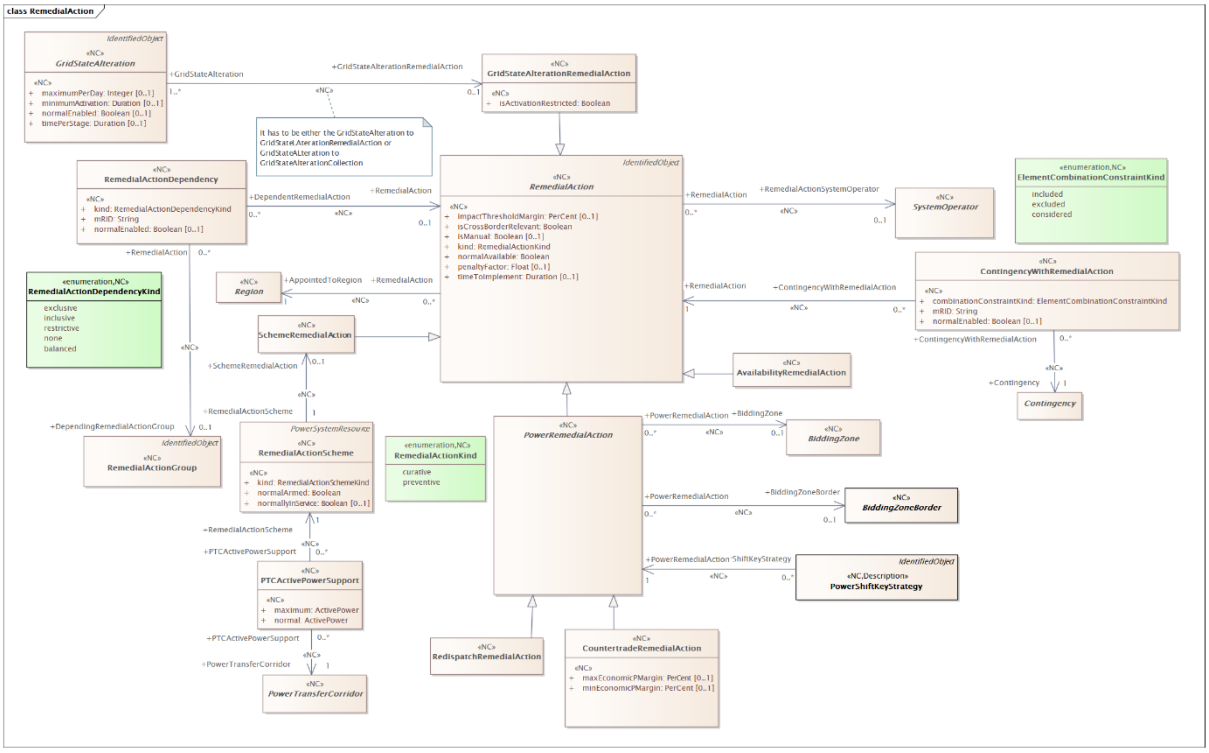
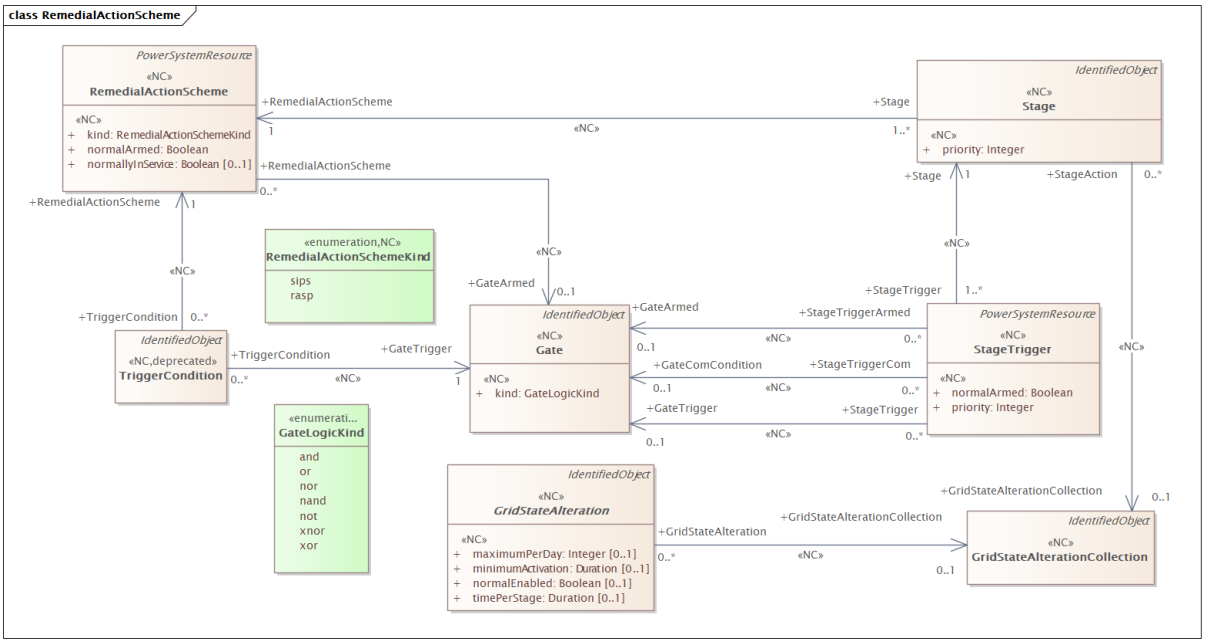


Figure 5 – Class diagram RemedialActionProfile::RemedialAction

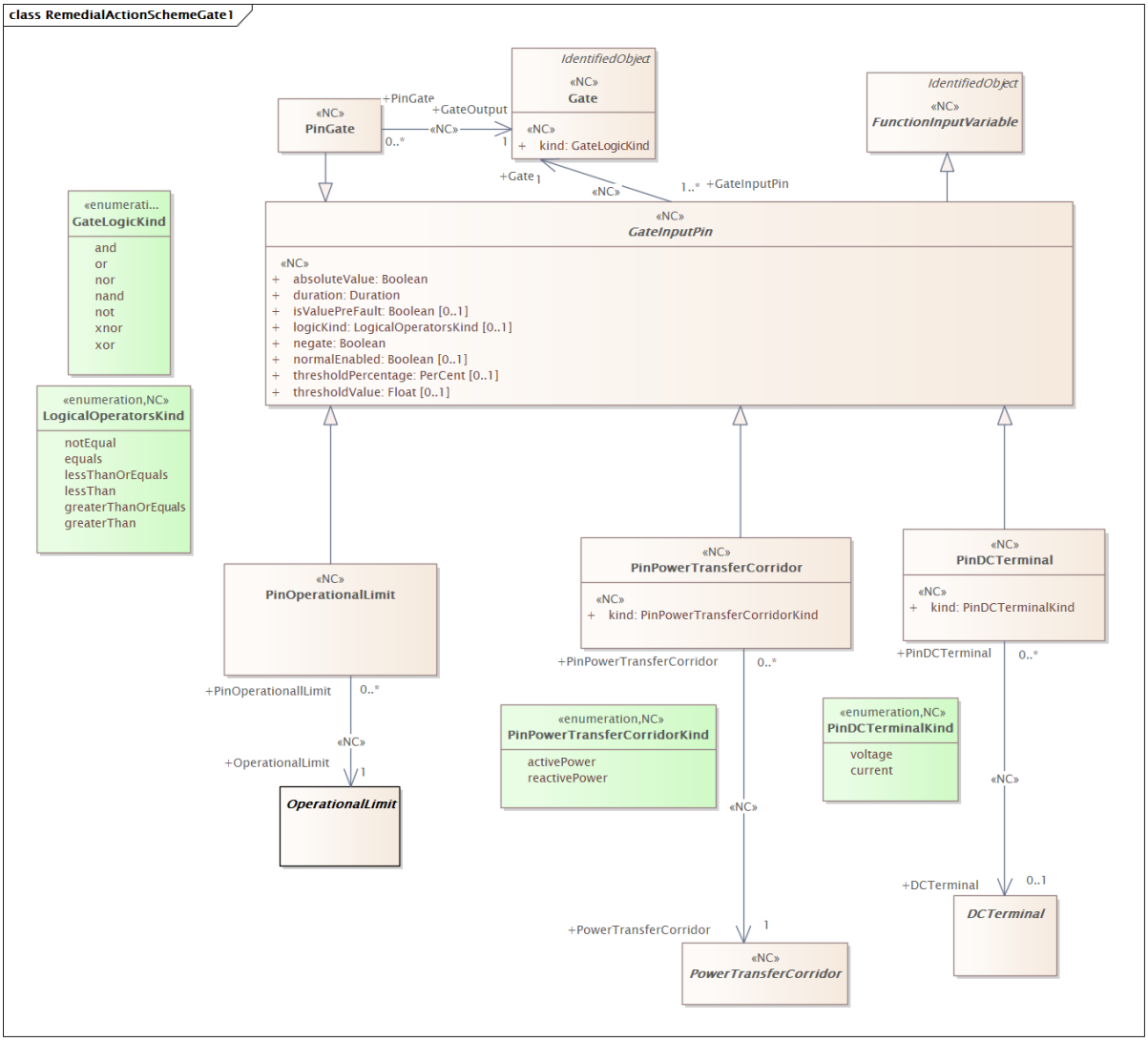
579 Figure 5: The diagram contains main classes related to the remedial action.



580

581 **Figure 6 – Class diagram RemedialActionProfile::RemedialActionScheme**

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



583

584 **Figure 7 – Class diagram RemedialActionProfile::RemedialActionSchemeGate1**

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

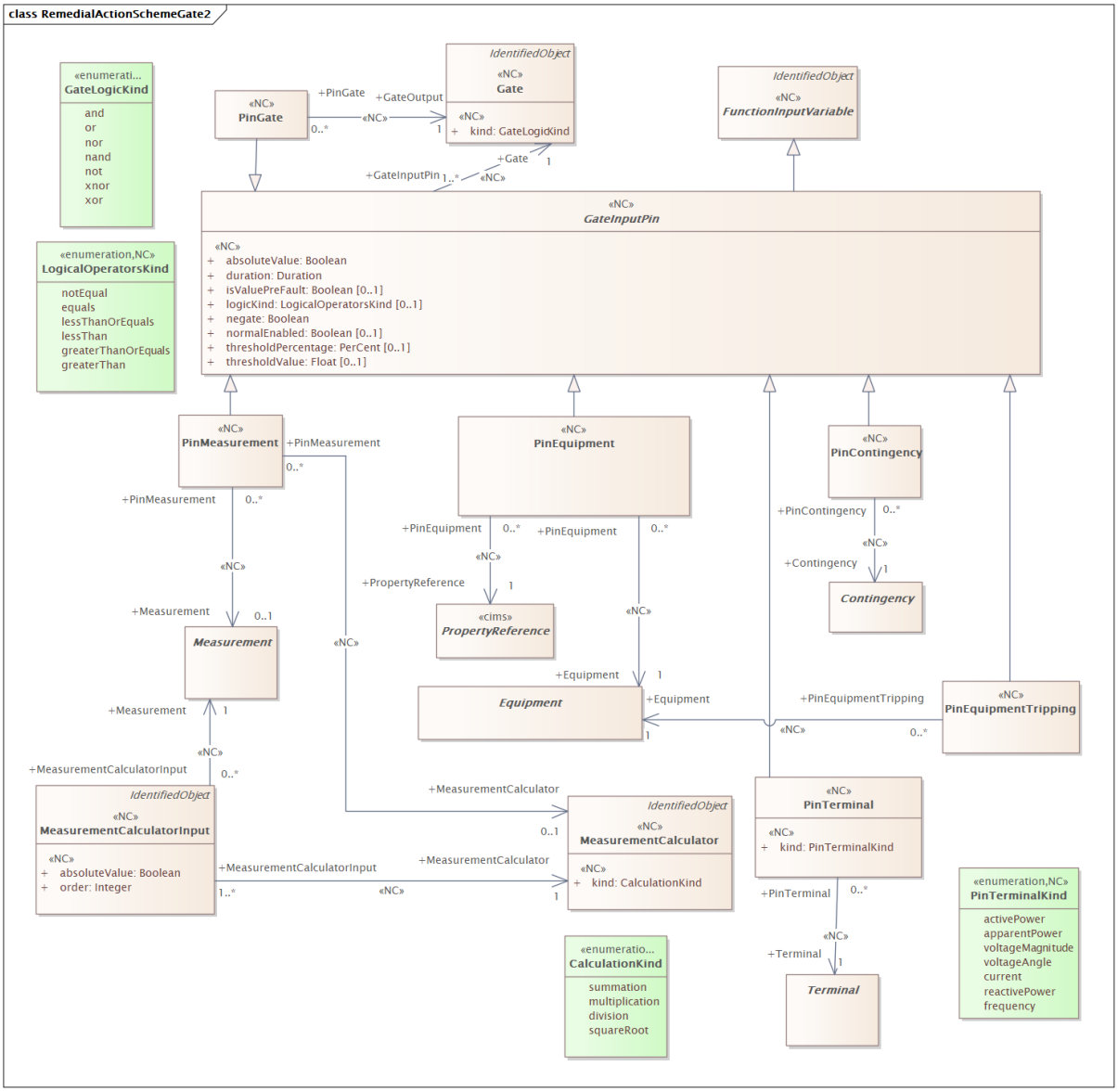


Figure 8 – Class diagram RemedialActionProfile::RemedialActionSchemeGate2

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

3.2 (abstract) ACDCConverter 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 (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 1 shows all attributes of ACDCTerminal.

598 **Table 1 – 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

599

600 ### 3.4 (abstract) BatteryUnit root class

601 An electrochemical energy storage device.

602 ### 3.5 (abstract) Contingency root class

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

604 ### 3.6 (abstract) DCTerminal root class

605 An electrical connection point to generic DC conducting equipment.

606 ### 3.7 (abstract) EnergyConsumer root class

607 Generic user of energy - a point of consumption on the power system model.
608 EnergyConsumer.pfixed, .qfixed, .pfixedPct and .qfixedPct have meaning only if there is no
609 LoadResponseCharacteristic associated with EnergyConsumer or if
610 LoadResponseCharacteristic.exponentModel is set to False.

611 ### 3.8 (abstract) EnergySource root class

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

613 ### 3.9 (abstract) Equipment root class

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

615 ### 3.10 (abstract) EquivalentInjection root class

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

618 ### 3.11 (abstract) ExternalNetworkInjection root class

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

620 ### 3.12 (abstract) IdentifiedObject root class

621 This is a root class to provide common identification for all classes needing identification and
622 naming attributes.
623 Table 2 shows all attributes of IdentifiedObject.

624 **Table 2 – 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	mult	type	description
name	0..1	String	The name is any free human readable and possibly non unique text naming the object.

3.13 (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.

3.14 (abstract) OperationalLimit root class

A value and normal value associated with a specific kind of limit.
The sub class value and normalValue attributes vary inversely to the associated OperationalLimitType.acceptableDuration (acceptableDuration for short).
If a particular piece of equipment has multiple operational limits of the same kind (apparent power, current, etc.), the limit with the greatest acceptableDuration shall have the smallest limit value and the limit with the smallest acceptableDuration shall have the largest limit value. Note: A large current can only be allowed to flow through a piece of equipment for a short duration without causing damage, but a lesser current can be allowed to flow for a longer duration.

3.15 (abstract) PowerElectronicsConnection root class

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

3.16 (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 3 shows all attributes of PowerSystemResource.

Table 3 – 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

3.17 (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.

3.18 (abstract) RotatingMachine root class

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

3.19 (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.

3.20 (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.

3.21 (abstract) Switch root class

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

3.22 (abstract) TapChanger root class

Mechanism for changing transformer winding tap positions.

3.23 (abstract) Terminal root class

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

3.24 (NC) CalculationKind enumeration

Kind of calculation operation that can be done to Measurement.
Table 4 shows all literals of CalculationKind.

Table 4 – 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).

3.25 Currency enumeration

Monetary currencies. ISO 4217 standard including 3-character currency code.
Table 5 shows all literals of Currency.

Table 5 – Literals of RemedialActionProfile::Currency

literal	value	description
AED		United Arab Emirates dirham.
AFN		Afghan afghani.
ALL		Albanian lek.
AMD		Armenian dram.
ANG		Netherlands Antillean guilder.
AOA		Angolan kwanza.
ARS		Argentine peso.
AUD		Australian dollar.
AWG		Aruban florin.
AZN		Azerbaijani manat.
BAM		Bosnia and Herzegovina convertible mark.
BBD		Barbados dollar.
BDT		Bangladeshi taka.
BGN		Bulgarian lev.
BHD		Bahraini dinar.
BIF		Burundian franc.
BMD		Bermudian dollar (customarily known as Bermuda dollar).
BND		Brunei dollar.
BOB		Boliviano.
BOV		Bolivian Mvdol (funds code).
BRL		Brazilian real.
BSD		Bahamian dollar.
BTN		Bhutanese ngultrum.

literal	value	description
BWP		Botswana pula.
BYR		Belarusian ruble.
BZD		Belize dollar.
CAD		Canadian dollar.
CDF		Congolese franc.
CHF		Swiss franc.
CLF		Unidad de Fomento (funds code), Chile.
CLP		Chilean peso.
CNY		Chinese yuan.
COP		Colombian peso.
COU		Unidad de Valor Real.
CRC		Costa Rican colon.
CUC		Cuban convertible peso.
CUP		Cuban peso.
CVE		Cape Verde escudo.
CZK		Czech koruna.
DJF		Djiboutian franc.
DKK		Danish krone.
DOP		Dominican peso.
DZD		Algerian dinar.
EEK		Estonian kroon.
EGP		Egyptian pound.
ERN		Eritrean nakfa.
ETB		Ethiopian birr.
EUR		Euro.
FJD		Fiji dollar.
FKP		Falkland Islands pound.
GBP		Pound sterling.
GEL		Georgian lari.
GHS		Ghanaian cedi.
GIP		Gibraltar pound.
GMD		Gambian dalasi.
GNF		Guinean franc.
GTQ		Guatemalan quetzal.
GYD		Guyanese dollar.
HKD		Hong Kong dollar.
HNL		Honduran lempira.
HRK		Croatian kuna.
HTG		Haitian gourde.
HUF		Hungarian forint.
IDR		Indonesian rupiah.

literal	value	description
ILS		Israeli new sheqel.
INR		Indian rupee.
IQD		Iraqi dinar.
IRR		Iranian rial.
ISK		Icelandic króna.
JMD		Jamaican dollar.
JOD		Jordanian dinar.
JPY		Japanese yen.
KES		Kenyan shilling.
KGS		Kyrgyzstani som.
KHR		Cambodian riel.
KMF		Comoro franc.
KPW		North Korean won.
KRW		South Korean won.
KWD		Kuwaiti dinar.
KYD		Cayman Islands dollar.
KZT		Kazakhstani tenge.
LAK		Lao kip.
LBP		Lebanese pound.
LKR		Sri Lanka rupee.
LRD		Liberian dollar.
LSL		Lesotho loti.
LTL		Lithuanian litas.
LVL		Latvian lats.
LYD		Libyan dinar.
MAD		Moroccan dirham.
MDL		Moldovan leu.
MGA		Malagasy ariary.
MKD		Macedonian denar.
MMK		Myanma kyat.
MNT		Mongolian tugrik.
MOP		Macanese pataca.
MRO		Mauritanian ouguiya.
MUR		Mauritian rupee.
MVR		Maldivian rufiyaa.
MWK		Malawian kwacha.
MXN		Mexican peso.
MYR		Malaysian ringgit.
MZN		Mozambican metical.
NAD		Namibian dollar.
NGN		Nigerian naira.

literal	value	description
NIO		Cordoba oro.
NOK		Norwegian krone.
NPR		Nepalese rupee.
NZD		New Zealand dollar.
OMR		Omani rial.
PAB		Panamanian balboa.
PEN		Peruvian nuevo sol.
PGK		Papua New Guinean kina.
PHP		Philippine peso.
PKR		Pakistani rupee.
PLN		Polish zloty.
PYG		Paraguayan guaraní.
QAR		Qatari rial.
RON		Romanian new leu.
RSD		Serbian dinar.
RUB		Russian rouble.
RWF		Rwandan franc.
SAR		Saudi riyal.
SBD		Solomon Islands dollar.
SCR		Seychelles rupee.
SDG		Sudanese pound.
SEK		Swedish krona/kronor.
SGD		Singapore dollar.
SHP		Saint Helena pound.
SLL		Sierra Leonean leone.
SOS		Somali shilling.
SRD		Surinamese dollar.
STD		São Tomé and Príncipe dobra.
SYP		Syrian pound.
SZL		Lilangeni.
THB		Thai baht.
TJS		Tajikistani somoni.
TMT		Turkmenistani manat.
TND		Tunisian dinar.
TOP		Tongan pa'anga.
TRY		Turkish lira.
TTD		Trinidad and Tobago dollar.
TWD		New Taiwan dollar.
TZS		Tanzanian shilling.
UAH		Ukrainian hryvnia.
UGX		Ugandan shilling.

literal	value	description
USD		United States dollar.
UYU		Uruguayan peso.
UZS		Uzbekistan som.
VEF		Venezuelan bolívar fuerte.
VND		Vietnamese Dong.
VUV		Vanuatu vatu.
WST		Samoan tala.
XAF		CFA franc BEAC.
XCD		East Caribbean dollar.
XOF		CFA Franc BCEAO.
XPF		CFP franc.
YER		Yemeni rial.
ZAR		South African rand.
ZMK		Zambian kwacha.
ZWL		Zimbabwe dollar.

3.26 (NC) ElementCombinationConstraintKind enumeration

Kind of constraint for an element combination.
Table 6 shows all literals of ElementCombinationConstraintKind.

Table 6 – Literals of RemedialActionProfile::ElementCombinationConstraintKind

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

3.27 (NC) GateLogicKind enumeration

Define the different logical operations.
Table 7 shows all literals of GateLogicKind.

Table 7 – 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.

literal	value	description
xor		A logical XOR operation. All input false or true will give false. Otherwise true.

3.28 (NC) LogicalOperatorsKind enumeration

Kinds of logical operators for comparison.
Table 8 shows all literals of LogicalOperatorsKind.

Table 8 – 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.

3.29 (NC) PinDCTerminalKind enumeration

The kind of quantities that can serve as an input value for the DCTerminal pin.
Table 9 shows all literals of PinDCTerminalKind.

Table 9 – Literals of RemedialActionProfile::PinDCTerminalKind

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

3.30 (NC) PinPowerTransferCorridorKind enumeration

The kind of quantities that can serve as an input value for the PowerTransferCorridor pin.
Table 10 shows all literals of PinPowerTransferCorridorKind.

Table 10 – Literals of RemedialActionProfile::PinPowerTransferCorridorKind

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

3.31 (NC) PinTerminalKind enumeration

The kind of quantities that can serve as an input value for the pin.
Table 11 shows all literals of PinTerminalKind.

Table 11 – 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.

literal	value	description
current		Current on the Terminal.
reactivePower		Reactive power on the Terminal.
frequency		Frequency on the Terminal.

3.32 (NC) RelativeDirectionKind enumeration

Kind of direction for the changes.
Table 12 shows all literals of RelativeDirectionKind.

Table 12 – 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.

3.33 (NC) RemedialActionDependencyKind enumeration

Kind of dependency between remedial actions.
Table 13 shows all literals of RemedialActionDependencyKind.

Table 13 – 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.
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.

3.34 (NC) RemedialActionKind enumeration

The different kinds for a remedial action.
Table 14 shows all literals of RemedialActionKind.

771 **Table 14 – 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.

772

773 ### 3.35 (NC) RemedialActionSchemeKind enumeration

774 Classification of Remedial Action Scheme.
775 Table 15 shows all literals of RemedialActionSchemeKind.

776 **Table 15 – 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.

777

778 ### 3.36 UnitMultiplier enumeration

779 The unit multipliers defined for the CIM. When applied to unit symbols, the unit symbol is
780 treated as a derived unit. Regardless of the contents of the unit symbol text, the unit symbol
781 shall be treated as if it were a single-character unit symbol. Unit symbols should not contain
782 multipliers, and it should be left to the multiplier to define the multiple for an entire data type.
783 For example, if a unit symbol is "m2Pers" and the multiplier is "k", then the value is k(m**2/s),
784 and the multiplier applies to the entire final value, not to any individual part of the value. This
785 can be conceptualized by substituting a derived unit symbol for the unit type. If one imagines
786 that the symbol "Þ" represents the derived unit "m2Pers", then applying the multiplier "k" can
787 be conceptualized simply as "kÞ".
788 For example, the SI unit for mass is "kg" and not "g". If the unit symbol is defined as "kg", then
789 the multiplier is applied to "kg" as a whole and does not replace the "k" in front of the "g". In
790 this case, the multiplier of "m" would be used with the unit symbol of "kg" to represent one gram.
791 As a text string, this violates the instructions in IEC 80000-1. However, because the unit symbol
792 in CIM is treated as a derived unit instead of as an SI unit, it makes more sense to conceptualize
793 the "kg" as if it were replaced by one of the proposed replacements for the SI mass symbol. If
794 one imagines that the "kg" were replaced by a symbol "Þ", then it is easier to conceptualize the
795 multiplier "m" as creating the proper unit "mÞ", and not the forbidden unit "mkg".
796 Table 16 shows all literals of UnitMultiplier.

797 **Table 16 – Literals of RemedialActionProfile::UnitMultiplier**

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

798

3.37 UnitSymbol enumeration

The derived units defined for usage in the CIM. In some cases, the derived unit is equal to an SI unit. Whenever possible, the standard derived symbol is used instead of the formula for the derived unit. For example, the unit symbol Farad is defined as "F" instead of "CPerV". In cases where a standard symbol does not exist for a derived unit, the formula for the unit is used as the unit symbol. For example, density does not have a standard symbol and so it is represented as "kgPerm3". With the exception of the "kg", which is an SI unit, the unit symbols do not contain multipliers and therefore represent the base derived unit to which a multiplier can be applied as a whole.
Every unit symbol is treated as an unparseable text as if it were a single-letter symbol. The meaning of each unit symbol is defined by the accompanying descriptive text and not by the text contents of the unit symbol.
To allow the widest possible range of serializations without requiring special character handling, several substitutions are made which deviate from the format described in IEC 80000-1. The division symbol "/" is replaced by the letters "Per". Exponents are written in plain text after the unit as "m3" instead of being formatted as "m" with a superscript of 3 or introducing a symbol as in "m^3". The degree symbol "°" is replaced with the letters "deg". Any clarification of the meaning for a substitution is included in the description for the unit symbol.
Non-SI units are included in list of unit symbols to allow sources of data to be correctly labelled with their non-SI units (for example, a GPS sensor that is reporting numbers that represent feet instead of meters). This allows software to use the unit symbol information correctly convert and scale the raw data of those sources into SI-based units.
The integer values are used for harmonization with IEC 61850.
Table 17 shows all literals of UnitSymbol.

Table 17 – Literals of RemedialActionProfile::UnitSymbol

literal	value	description
none		Dimension less quantity, e.g. count, per unit, etc.
s		Time in seconds.
W		Real power in watts (J/s). Electrical power may have real and reactive components. The real portion of electrical power (I^2R or VIcos(phi)), is expressed in Watts. See also apparent power and reactive power.
VAr		Reactive power in volt amperes reactive. The “reactive” or “imaginary” component of electrical power (VIsin(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.

3.38 (NC) ValueOffsetKind enumeration

The kind of the value offset.
Table 18 shows all literals of ValueOffsetKind.

Table 18 – 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.

literal	value	description
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.

3.39 ActivePower datatype

Product of RMS value of the voltage and the RMS value of the in-phase component of the current.
Table 19 shows all attributes of ActivePower.

Table 19 – Attributes of RemedialActionProfile::ActivePower

name	mult	type	description
multiplier	0..1	UnitMultiplier	
unit	0..1	UnitSymbol	
value	0..1	Float	

3.40 PerCent datatype

Percentage on a defined base. For example, specify as 100 to indicate at the defined base.
Table 20 shows all attributes of PerCent.

Table 20 – Attributes of RemedialActionProfile::PerCent

name	mult	type	description
value	0..1	Float	Normally 0 to 100 on a defined base.
unit	0..1	UnitSymbol	
multiplier	0..1	UnitMultiplier	

3.41 ReactivePower datatype

Product of RMS value of the voltage and the RMS value of the quadrature component of the current.
Table 21 shows all attributes of ReactivePower.

Table 21 – Attributes of RemedialActionProfile::ReactivePower

name	mult	type	description
value	0..1	Float	
unit	0..1	UnitSymbol	
multiplier	0..1	UnitMultiplier	

3.42 Seconds datatype

Time, in seconds.
Table 22 shows all attributes of Seconds.

Table 22 – Attributes of RemedialActionProfile::Seconds

name	mult	type	description
value	0..1	Float	Time, in seconds
unit	0..1	UnitSymbol	
multiplier	0..1	UnitMultiplier	

3.43 Boolean primitive

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

3.44 Duration primitive

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

3.45 Float primitive

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

3.46 Integer primitive

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

3.47 String primitive

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

3.48 (NC) ACDCConverterAction

Inheritance path = SetPointAction : GridStateAlteration : IdentifiedObject
Alternate current Direct current (ACDC) converter action.
Table 23 shows all attributes of ACDCConverterAction.

Table 23 – Attributes of RemedialActionProfile::ACDCConverterAction

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 24 shows all association ends of ACDCConverterAction with other classes.

Table 24 – 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

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

3.49 (NC) AvailabilityRemedialAction

Inheritance path = RemedialAction : IdentifiedObject

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

Table 25 shows all attributes of AvailabilityRemedialAction.

Table 25 – 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 26 shows all association ends of AvailabilityRemedialAction with other classes.

Table 26 – 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

3.50 (NC) BatteryUnitAction

Inheritance path = SetPointAction : GridStateAlteration : IdentifiedObject

Battery unit setpoint action.

Table 27 shows all attributes of BatteryUnitAction.

Table 27 – 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

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

Table 28 shows all association ends of BatteryUnitAction with other classes.

Table 28 – 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

3.51 (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.

3.52 (NC) BiddingZoneAction

Inheritance path = SetPointAction : GridStateAlteration : IdentifiedObject
Bidding zone set point action.
Table 29 shows all attributes of BiddingZoneAction.

Table 29 – 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

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

Table 30 – 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

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

3.53 (abstract,NC) BiddingZoneBorder root class

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

3.54 (NC) ContingencyWithRemedialAction root class

Combination of a contingency and a remedial action. ContingencyWithRemedialAction shall not be instantiated for preventive RemedialAction (RemedialAction.kind equals RemedialActionKind.preventive).
Table 31 shows all attributes of ContingencyWithRemedialAction.

Table 31 – 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 32 shows all association ends of ContingencyWithRemedialAction with other classes.

Table 32 – 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.

3.55 (abstract,NC) ControlFunctionBlock root class

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

3.56 (NC) ControlFunctionBlockAction

Inheritance path = SetPointAction : GridStateAlteration : IdentifiedObject
Action for setting the control function block target values.
Table 33 shows all attributes of ControlFunctionBlockAction.

Table 33 – 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 34 shows all association ends of ControlFunctionBlockAction with other classes.

Table 34 – 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

3.57 (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 35 shows all attributes of CountertradeRemedialAction.

Table 35 – 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].

name	mult	type	description
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

Table 36 shows all association ends of CountertradeRemedialAction with other classes.

Table 36 – Association ends of RemedialActionProfile::CountertradeRemedialAction 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

3.58 (NC) EnergySourceModification

Inheritance path = SetPointAction : GridStateAlteration : IdentifiedObject

Energy source action.

Table 37 shows all attributes of EnergySourceModification.

Table 37 – 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 38 shows all association ends of EnergySourceModification with other classes.

Table 38 – 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.
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

3.59 (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.

3.60 (NC) EquipmentControllerAction

Inheritance path = SetPointAction : GridStateAlteration : IdentifiedObject
Action for setting the equipment controller action.
Table 39 shows all attributes of EquipmentControllerAction.

Table 39 – 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 40 shows all association ends of EquipmentControllerAction with other classes.

Table 40 – 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

3.61 (NC) EquivalentInjectionAction

Inheritance path = SetPointAction : GridStateAlteration : IdentifiedObject

Equivalent injection action.
Table 41 shows all attributes of EquivalentInjectionAction.

Table 41 – 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 42 shows all association ends of EquivalentInjectionAction with other classes.

Table 42 – 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

3.62 (NC) ExternalNetworkInjectionAction

Inheritance path = SetPointAction : GridStateAlteration : IdentifiedObject
External network injection action.
Table 43 shows all attributes of ExternalNetworkInjectionAction.

Table 43 – 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 44 shows all association ends of ExternalNetworkInjectionAction with other classes.

Table 44 – 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

3.63 (abstract,NC) FunctionInputVariable

Inheritance path = IdentifiedObject
Functional input variable defines the domain of the function.
Table 45 shows all attributes of FunctionInputVariable.

Table 45 – 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

3.64 (NC) Gate

Inheritance path = IdentifiedObject
Logical gate that supports a logical operation based on the input.
Table 46 shows all attributes of Gate.

Table 46 – 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

3.65 (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 47 shows all attributes of GateInputPin.

Table 47 – 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).
duration	1..1	Duration	(NC) The time duration for which the condition is satisfied before acting. Default is 0 seconds.

name	mult	type	description
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.
normalEnabled	0..1	Boolean	(NC) True, if the gate input pin is normally enabled (active). Otherwise false.
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 48 shows all association ends of GateInputPin with other classes.

Table 48 – 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.

3.66 (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 49 shows all attributes of GridStateAlteration.

Table 49 – Attributes of RemedialActionProfile::GridStateAlteration

name	mult	type	description
normalEnabled	0..1	Boolean	(NC) It identifies if the element is normally enabled. True means normally enabled, False means not normally enabled.
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 50 shows all association ends of GridStateAlteration with other classes.

Table 50 – 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.

3.67 (NC) GridStateAlterationCollection

Inheritance path = IdentifiedObject
A collection of grid state alterations.
Table 51 shows all attributes of GridStateAlterationCollection.

Table 51 – 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

3.68 (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 52 shows all attributes of GridStateAlterationRemedialAction.

Table 52 – 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 or 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 53 shows all association ends of GridStateAlterationRemedialAction with other classes.

Table 53 – 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

3.69 (NC) InServiceAction

Inheritance path = GridStateAlteration : IdentifiedObject

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

Table 54 shows all attributes of InServiceAction.

Table 54 – 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

Table 55 shows all association ends of InServiceAction with other classes.

Table 55 – Association ends of RemedialActionProfile::InServiceAction with other 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

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

3.70 (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 56 shows all attributes of IntertemporalPropertyRange.

Table 56 – 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 57 shows all association ends of IntertemporalPropertyRange with other classes.

Table 57 – Association ends of RemedialActionProfile::IntertemporalPropertyRange with other classes

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

3.71 (NC) LoadAction

Inheritance path = SetPointAction : GridStateAlteration : IdentifiedObject
Load action.
Table 58 shows all attributes of LoadAction.

Table 58 – 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	mult	type	description
name	0..1	String	inherited from: IdentifiedObject

Table 59 shows all association ends of LoadAction with other classes.

Table 59 – 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
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

3.72 (NC) MeasurementCalculator

Inheritance path = IdentifiedObject
Result of a calculation of one or more measurement.
Table 60 shows all attributes of MeasurementCalculator.

Table 60 – 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

3.73 (NC) MeasurementCalculatorInput

Inheritance path = IdentifiedObject
Input to measurement calculation. It supports Analog, Discrete and Accumulator measurements.
Table 61 shows all attributes of MeasurementCalculatorInput.

Table 61 – 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).
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 62 shows all association ends of MeasurementCalculatorInput with other classes.

Table 62 – 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.

3.74 (NC) PinContingency

Inheritance path = GateInputPin : FunctionInputVariable : IdentifiedObject
Input pin associated with a Contingency. It is used for comparison.
Table 63 shows all attributes of PinContingency.

Table 63 – 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
normalEnabled	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 64 shows all association ends of PinContingency with other classes.

Table 64 – 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

3.75 (NC) PinDCTerminal

Inheritance path = GateInputPin : FunctionInputVariable : IdentifiedObject
Input pin associated with a DCTerminal. It is used for comparison.
Table 65 shows all attributes of PinDCTerminal.

Table 65 – Attributes of RemedialActionProfile::PinDCTerminal

name	mult	type	description
kind	1..1	PinDCTerminalKind	(NC) The kind of quantity which is used as an input value.

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
normalEnabled	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 66 shows all association ends of PinDCTerminal with other classes.

Table 66 – 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

3.76 (NC) PinEquipment

Inheritance path = GateInputPin : FunctionInputVariable : IdentifiedObject
Input pin associated with an Equipment. It is used for the comparison.
Table 67 shows all attributes of PinEquipment.

Table 67 – 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
normalEnabled	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 68 shows all association ends of PinEquipment with other classes.

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

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

3.77 (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 69 shows all attributes of PinEquipmentTripping.

Table 69 – 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
normalEnabled	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 70 shows all association ends of PinEquipmentTripping with other classes.

Table 70 – 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

3.78 (NC) PinGate

Inheritance path = GateInputPin : FunctionInputVariable : IdentifiedObject

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

Table 71 shows all attributes of PinGate.

Table 71 – Attributes of RemedialActionProfile::PinGate

name	mult	type	description
absoluteValue	1..1	Boolean	(NC) inherited from: GateInputPin

name	mult	type	description
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
normalEnabled	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 72 shows all association ends of PinGate with other classes.

Table 72 – 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

3.79 (NC) PinMeasurement

Inheritance path = GateInputPin : FunctionInputVariable : IdentifiedObject
Input pin associated with a Measurement. It is used for comparison.
Table 73 shows all attributes of PinMeasurement.

Table 73 – 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
normalEnabled	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 74 shows all association ends of PinMeasurement with other classes.

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

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

3.80 (NC) PinOperationalLimit

Inheritance path = GateInputPin : FunctionInputVariable : IdentifiedObject
Input pin associated with the limits of a Terminal. It is used for comparison.
Table 75 shows all attributes of PinOperationalLimit.

Table 75 – 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
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
normalEnabled	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 76 shows all association ends of PinOperationalLimit with other classes.

Table 76 – 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

3.81 (NC) PinPowerTransferCorridor

Inheritance path = GateInputPin : FunctionInputVariable : IdentifiedObject
Input pin associated with a PowerTransferCorridor. It is used for comparison.
Table 77 shows all attributes of PinPowerTransferCorridor.

Table 77 – 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
normalEnabled	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 78 shows all association ends of PinPowerTransferCorridor with other classes.

Table 78 – 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

3.82 (NC) PinTerminal

Inheritance path = GateInputPin : FunctionInputVariable : IdentifiedObject

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

Table 79 shows all attributes of PinTerminal.

Table 79 – 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
normalEnabled	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 80 shows all association ends of PinTerminal with other classes.

Table 80 – 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

3.83 (NC) PowerElectronicsConnectionAction

Inheritance path = SetPointAction : GridStateAlteration : IdentifiedObject
Power electronics setpoint action.
Table 81 shows all attributes of PowerElectronicsConnectionAction.

Table 81 – 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 82 shows all association ends of PowerElectronicsConnectionAction with other classes.

Table 82 – 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
0..*	GridStateAlterationCollection	0..1	GridStateAlterationCollection	(NC) inherited from: GridStateAlteration
0..*	PropertyReference	1..1	PropertyReference	(NC) inherited from: GridStateAlteration

3.84 (abstract,NC) PowerRemedialAction

Inheritance path = RemedialAction : IdentifiedObject
Energy remedial action describes actions to rearrange power schedules.
Table 83 shows all attributes of PowerRemedialAction.

Table 83 – 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

Table 84 shows all association ends of PowerRemedialAction with other classes.

Table 84 – 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

3.85 (NC,Description) PowerShiftKeyStrategy

Inheritance path = IdentifiedObject
Strategy of the power shift key.
Table 85 shows all attributes of PowerShiftKeyStrategy.

Table 85 – 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

Table 86 shows all association ends of PowerShiftKeyStrategy with other classes.

Table 86 – Association ends of RemedialActionProfile::PowerShiftKeyStrategy with other classes

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

3.86 (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.

3.87 (abstract,cims) PropertyReference root class

The reference to a class and one of its properties.

3.88 (NC) PTCActivePowerSupport root class

Defines the active power capability (support) of the scheme in relation to a PowerTransferCorridor.
Table 87 shows all attributes of PTCActivePowerSupport.

Table 87 – 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 88 shows all association ends of PTCActivePowerSupport with other classes.

Table 88 – 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.
0..*	RemedialActionScheme	1..1	RemedialActionScheme	(NC) The RemedialActionScheme which has active power support from the PowerTransferCorridor.

3.89 (abstract,NC) RangeConstraint

Inheritance path = IdentifiedObject
Defines the rage constraint.
Table 89 shows all attributes of RangeConstraint.

Table 89 – 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 1.0 equals true and 0.0 equals false. In case of integer the 0 should be included as a decimal, e.g. 2.0.

name	mult	type	description
			If the valueKind equals incremental or incrementalPercentage, then the attribute value shall be positive (greater than zero). If the valueKind equals absolute, it does not constraint attribute value to be positive. If the valueKind equals incrementalPercentage, then the attribute value “100.0” shall represent 100%. The attribute value shall be positive float (0.0 value is not allowed) and can exceed 100.0.
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

Table 90 shows all association ends of RangeConstraint with other classes.

Table 90 – 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.

3.90 (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 91 shows all attributes of RedispatchRemedialAction.

Table 91 – 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 92 shows all association ends of RedispatchRemedialAction with other classes.

Table 92 – Association ends of RemedialActionProfile::RedispatchRemedialAction with other classes

mult from	name	mult to	type	description
0..*	BiddingZone	0..1	BiddingZone	(NC) inherited from: PowerRemedialAction

mult from	name	mult to	type	description
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

3.91 (abstract,NC) Region root class

A region where the system operator belongs to.

3.92 (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 93 shows all attributes of RegulatingControlAction.

Table 93 – 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 94 shows all association ends of RegulatingControlAction with other classes.

Table 94 – 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

3.93 (abstract,NC) RemedialAction

Inheritance path = IdentifiedObject
Remedial action describes one or more actions that can be performed on a given power system model situation to eliminate one or more identified breaches of constraints. The remedial action can be costly, and have a cost characteristic, or non costly.
Table 95 shows all attributes of RemedialAction.

Table 95 – 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 – 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

Table 96 shows all association ends of RemedialAction with other classes.

Table 96 – 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.

3.94 (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 97 shows all attributes of RemedialActionDependency.

Table 97 – 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.

Table 98 shows all association ends of RemedialActionDependency with other classes.

Table 98 – 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.

3.95 (NC) RemedialActionGroup

Inheritance path = IdentifiedObject

Grouping of remedial actions that can be operated together.

Table 99 shows all attributes of RemedialActionGroup.

Table 99 – 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

3.96 (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 100 shows all attributes of RemedialActionScheme.

Table 100 – 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 101 shows all association ends of RemedialActionScheme with other classes.

Table 101 – 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.

3.97 (NC) RotatingMachineAction

Inheritance path = SetPointAction : GridStateAlteration : IdentifiedObject

Rotating machine action.

Table 102 shows all attributes of RotatingMachineAction.

Table 102 – 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	mult	type	description
name	0..1	String	inherited from: IdentifiedObject

Table 103 shows all association ends of RotatingMachineAction with other classes.

Table 103 – Association ends of RemedialActionProfile::RotatingMachineAction with 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

3.98 (NC) SchemeRemedialAction

Inheritance path = RemedialAction : IdentifiedObject
Scheme remedial action is remedial action that involves a scheme that can include conditional logic and stages of grid alteration. The primary remedial action is the arming of these schemes, that will then perform curative remedial action when the condition is met. System Integrity Protection Scheme (SIPS) and Special Protection Scheme (SPS) are example of this.
Table 104 shows all attributes of SchemeRemedialAction.

Table 104 – 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

Table 105 shows all association ends of SchemeRemedialAction with other classes.

Table 105 – 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

3.99 (abstract,NC) SetPointAction

Inheritance path = GridStateAlteration : IdentifiedObject
Setpoint action.
Table 106 shows all attributes of SetPointAction.

Table 106 – 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 107 shows all association ends of SetPointAction with other classes.

Table 107 – 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

3.100 (NC) ShuntCompensatorModification

Inheritance path = SetPointAction : GridStateAlteration : IdentifiedObject
Shunt compensator action.
Table 108 shows all attributes of ShuntCompensatorModification.

Table 108 – Attributes of RemedialActionProfile::ShuntCompensatorModification

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 109 shows all association ends of ShuntCompensatorModification with other classes.

Table 109 – 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

3.101 (NC) Stage

Inheritance path = IdentifiedObject
Stage of a remedial action scheme.
Table 110 shows all attributes of Stage.

Table 110 – 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 111 shows all association ends of Stage with other classes.

Table 111 – 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.

3.102 (NC) StageTrigger

Inheritance path = PowerSystemResource : IdentifiedObject
Stage that is triggered either by TriggerCondition or by gate condition within a stage.
Table 112 shows all attributes of StageTrigger.

Table 112 – 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

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

Table 113 shows all association ends of StageTrigger with other classes.

Table 113 – 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.

3.103 (NC) StaticPropertyRange

Inheritance path = RangeConstraint : IdentifiedObject
Defines the static range, which means that this is the minimum and/or maximum of an attribute value. The value provided by the schedule replaces the value of the attribute to which the schedule refers to.
In case that the PropertyReference refers to Boolean type attributes, RangeConstraint.direction shall be none or upAndDown and the RangeConstraint.valueKind shall be absolute. If the direction is none then optimization of the attribute referenced by the PropertyReference is not possible if the current status is already as the value in the range. Otherwise if the direction is upAndDown, the optimization can change from true to false or vice versa independently of the initial value in the operational scenario.
For instance for a tap changer related grid state alteration for a particular point in time, if the range of TapChanger.step is to be restricted, the value of the schedule will represent that new TapChanger.step range.
Table 114 shows all attributes of StaticPropertyRange.

Table 114 – 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	0..1	String	inherited from: IdentifiedObject

Table 115 shows all association ends of StaticPropertyRange with other classes.

Table 115 – Association ends of RemedialActionProfile::StaticPropertyRange with other classes

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

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

3.104 (NC) StaticVarCompensatorAction

Inheritance path = SetPointAction : GridStateAlteration : IdentifiedObject

Static Var compensator action.

Table 116 shows all attributes of StaticVarCompensatorAction.

Table 116 – 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 117 shows all association ends of StaticVarCompensatorAction with other classes.

Table 117 – 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

3.105 (Description) Substation root class

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

3.106 (abstract,NC) SystemOperator root class

A power system operation post responsible for performing and coordinating the real-time and planning functions required to maintain the secure, reliable, and efficient operation of the power system within a defined operational scope.

Note: SystemOperator represents the execution of power system operational responsibilities in practice. While the operational post exists independently at the ecosystem level, SystemOperator identifies the specific agent fulfilling that responsibility at a given time. The operational scope may correspond to a control area, network, voltage domain, or other defined boundary.

3.107 (NC) TapPositionAction

Inheritance path = GridStateAlteration : IdentifiedObject

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

Table 118 shows all attributes of TapPositionAction.

Table 118 – 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

Table 119 shows all association ends of TapPositionAction with other classes.

Table 119 – Association ends of RemedialActionProfile::TapPositionAction with other 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

3.108 (NC) TopologyAction

Inheritance path = GridStateAlteration : IdentifiedObject

Topology action means the connection or disconnection of a switch in the grid model compared to the base case.

Table 120 shows all attributes of TopologyAction.

Table 120 – 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

Table 121 shows all association ends of TopologyAction with other classes.

Table 121 – Association ends of RemedialActionProfile::TopologyAction with other 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

3.109 (deprecated,NC) TriggerCondition

Inheritance path = IdentifiedObject

The condition that triggers a remedial action scheme.

Table 122 shows all attributes of TriggerCondition.

Table 122 – 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

Table 123 shows all association ends of TriggerCondition with other classes.

Table 123 – Association ends of RemedialActionProfile::TriggerCondition with other classes

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

Annex A (informative): Sample data

A.1 General

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

A.2 Sample instance data

Test data files are available in the CIM EG SharePoint.