



European Network of
Transmission System Operators
for Electricity

REMEDIAL ACTION SCHEDULE PROFILE SPECIFICATION

2024-10-16

ICTC APPROVED
VERSION 2.3.1

1 Copyright notice:

2 **Copyright © ENTSO-E. All Rights Reserved.**

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

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

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

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

17 **NOTE CONCERNING WORDING USED IN THIS DOCUMENT**

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

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

32

Revision History

Version	Date	Paragraph	Comments
1.0.0	2021-03-22		Document for SOC approval
2.0.0	2021-10-12		For CIM EG review. Association to OverlappingZone was added to the profile.
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.
2.3.1-alpha	2024-09-07		For CIM WG review.

34	CONTENTS		
35	Copyright notice:.....		2
36	Revision History.....		3
37	CONTENTS		4
38	1 Introduction		8
39	2 Application profile specification		8
40	2.1 Version information		8
41	2.2 Constraints naming convention		8
42	2.3 Profile constraints		9
43	2.4 Metadata.....		12
44	2.4.1 Constraints		12
45	2.4.2 Reference metadata		12
46	3 Detailed Profile Specification		13
47	3.1 General.....		13
48	3.2 (abstract,NC) BaseIrregularTimeSeries		13
49	3.3 (abstract,NC) BaseTimeSeries		13
50	3.4 (abstract) Contingency root class		14
51	3.5 (NC) EventSchedule		14
52	3.6 (NC) EventTimePoint root class		14
53	3.7 (abstract,NC) GenericValueSchedule		15
54	3.8 (NC) GenericValueTimePoint root class		15
55	3.9 (abstract,NC) GridStateAlteration		16
56	3.10 (NC) GridStateIntensitySchedule		16
57	3.11 (abstract) IdentifiedObject root class		17
58	3.12 (abstract,NC) RemedialAction		17
59	3.13 (NC) RemedialActionCost root class		17
60	3.14 (NC) RemedialActionSchedule		18
61	3.15 (NC) RemedialActionScheduleAcceptance root class		19
62	3.16 (abstract,NC) Region root class		19
63	3.17 (abstract,NC) ScheduleResource root class		20
64	3.18 (abstract,NC) SecurityCoordinator root class		20
65	3.19 (abstract,NC) SystemOperator root class		20
66	3.20 (NC) CostSettledKind enumeration		20
67	3.21 (NC) RemedialActionScheduleAcceptanceKind enumeration		20
68	3.22 (NC) TimeSeriesInterpolationKind enumeration		20
69	3.23 (NC) RemedialActionScheduleStatusKind enumeration		21
70	3.24 UnitMultiplier enumeration		21
71	3.25 UnitSymbol enumeration		21
72	3.26 (NC) ValueOffsetKind enumeration		22
73	3.27 ActivePower datatype		23
74	3.28 RealEnergy datatype.....		23
75	3.29 Seconds datatype		23
76	3.30 Boolean primitive		23
77	3.31 DateTime primitive		23

78	3.32	Duration primitive.....	23
79	3.33	Decimal primitive	24
80	3.34	Float primitive	24
81	3.35	String primitive.....	24
82	3.36	(NC,Description) RemedialActionSchemeSchedule	24
83	3.37	(NC) PowerSchedule	24
84	3.38	(NC) PowerTimePoint root class	24
85	3.39	ReactivePower datatype	25
86	3.40	Integer primitive	25
87	3.41	(abstract,NC) PowerBidSchedule	25
88	3.42	(abstract,NC) AvailabilityRemedialAction	26
89	3.43	(NC,Description) SchemeRemedialAction	26
90	3.44	(NC,Description) AvailabilitySchedule	26
91	3.45	Currency enumeration.....	27
92	3.46	(NC) ProposingRemedialActionScheduleShare root class	31
93	3.47	(NC) RemedialActionScheduleGroup	32
94	3.48	(abstract,NC) PowerScheduleAction	32
95	3.49	(NC) CountertradeScheduleAction	32
96	3.50	(NC) RedispatchScheduleAction	33
97	3.51	(NC) RemedialActionScheduleDependency root class.....	34
98	3.52	(NC) RemedialActionScheduleDependencyKind enumeration.....	34
99		Annex A (informative): Sample data	36
100	A.1	General.....	36
101	A.2	Sample instance data.....	36
102			
103		List of figures	
104		Figure 1 – Other diagram	
105		RemedialActionScheduleProfile::RemedialActionScheduleProfile	13
106			
107		List of tables	
108		Table 1 – Attributes of RemedialActionScheduleProfile::BaselrregularTimeSeries	13
109		Table 2 – Attributes of RemedialActionScheduleProfile::BaseTimeSeries	14
110		Table 3 – Attributes of RemedialActionScheduleProfile::EventSchedule.....	14
111		Table 4 – Association ends of RemedialActionScheduleProfile::EventSchedule with	
112		other classes	14
113		Table 5 – Attributes of RemedialActionScheduleProfile::EventTimePoint	14
114		Table 6 – Association ends of RemedialActionScheduleProfile::EventTimePoint with	
115		other classes	15
116		Table 7 – Attributes of RemedialActionScheduleProfile::GenericValueSchedule	15
117		Table 8 – Association ends of RemedialActionScheduleProfile::GenericValueSchedule	
118		with other classes	15
119		Table 9 – Attributes of RemedialActionScheduleProfile::GenericValueTimePoint	15

120	Table 10 – Association ends of	
121	RemedialActionScheduleProfile::GenericValueTimePoint with other classes	16
122	Table 11 – Attributes of RemedialActionScheduleProfile::GridStateAlteration	16
123	Table 12 – Attributes of RemedialActionScheduleProfile::GridStateIntensitySchedule	16
124	Table 13 – Association ends of	
125	RemedialActionScheduleProfile::GridStateIntensitySchedule with other classes	16
126	Table 14 – Attributes of RemedialActionScheduleProfile::IdentifiedObject	17
127	Table 15 – Attributes of RemedialActionScheduleProfile::RemedialAction	17
128	Table 16 – Attributes of RemedialActionScheduleProfile::RemedialActionCost	17
129	Table 17 – Association ends of RemedialActionScheduleProfile::RemedialActionCost	
130	with other classes	18
131	Table 18 – Attributes of RemedialActionScheduleProfile::RemedialActionSchedule	18
132	Table 19 – Association ends of	
133	RemedialActionScheduleProfile::RemedialActionSchedule with other classes	19
134	Table 20 – Attributes of	
135	RemedialActionScheduleProfile::RemedialActionScheduleAcceptance	19
136	Table 21 – Association ends of	
137	RemedialActionScheduleProfile::RemedialActionScheduleAcceptance with other	
138	classes	19
139	Table 22 – Literals of RemedialActionScheduleProfile::CostSettledKind	20
140	Table 23 – Literals of	
141	RemedialActionScheduleProfile::RemedialActionScheduleAcceptanceKind	20
142	Table 24 – Literals of RemedialActionScheduleProfile::TimeSeriesInterpolationKind	20
143	Table 25 – Literals of	
144	RemedialActionScheduleProfile::RemedialActionScheduleStatusKind	21
145	Table 26 – Literals of RemedialActionScheduleProfile::UnitMultiplier	21
146	Table 27 – Literals of RemedialActionScheduleProfile::UnitSymbol	22
147	Table 28 – Literals of RemedialActionScheduleProfile::ValueOffsetKind	22
148	Table 29 – Attributes of RemedialActionScheduleProfile::ActivePower	23
149	Table 30 – Attributes of RemedialActionScheduleProfile::RealEnergy	23
150	Table 31 – Attributes of RemedialActionScheduleProfile::Seconds	23
151	Table 32 – Attributes of	
152	RemedialActionScheduleProfile::RemedialActionSchemeSchedule	24
153	Table 33 – Association ends of	
154	RemedialActionScheduleProfile::RemedialActionSchemeSchedule with other classes	24
155	Table 34 – Attributes of RemedialActionScheduleProfile::PowerSchedule	24
156	Table 35 – Attributes of RemedialActionScheduleProfile::PowerTimePoint	25
157	Table 36 – Association ends of RemedialActionScheduleProfile::PowerTimePoint with	
158	other classes	25
159	Table 37 – Attributes of RemedialActionScheduleProfile::ReactivePower	25
160	Table 38 – Attributes of RemedialActionScheduleProfile::PowerBidSchedule	26
161	Table 39 – Attributes of RemedialActionScheduleProfile::AvailabilityRemedialAction	26
162	Table 40 – Attributes of RemedialActionScheduleProfile::SchemeRemedialAction	26
163	Table 41 – Attributes of RemedialActionScheduleProfile::AvailabilitySchedule	26

164	Table 42 – Association ends of RemedialActionScheduleProfile::AvailabilitySchedule	
165	with other classes	27
166	Table 43 – Literals of RemedialActionScheduleProfile::Currency	27
167	Table 44 – Attributes of	
168	RemedialActionScheduleProfile::ProposingRemedialActionScheduleShare	31
169	Table 45 – Association ends of	
170	RemedialActionScheduleProfile::ProposingRemedialActionScheduleShare with other	
171	classes	31
172	Table 46 – Attributes of	
173	RemedialActionScheduleProfile::RemedialActionScheduleGroup	32
174	Table 47 – Attributes of RemedialActionScheduleProfile::PowerScheduleAction	32
175	Table 48 – Association ends of RemedialActionScheduleProfile::PowerScheduleAction	
176	with other classes	32
177	Table 49 – Attributes of RemedialActionScheduleProfile::CountertradeScheduleAction	33
178	Table 50 – Association ends of	
179	RemedialActionScheduleProfile::CountertradeScheduleAction with other classes	33
180	Table 51 – Attributes of RemedialActionScheduleProfile::RedispatchScheduleAction	33
181	Table 52 – Association ends of	
182	RemedialActionScheduleProfile::RedispatchScheduleAction with other classes	33
183	Table 53 – Attributes of	
184	RemedialActionScheduleProfile::RemedialActionScheduleDependency	34
185	Table 54 – Association ends of	
186	RemedialActionScheduleProfile::RemedialActionScheduleDependency with other	
187	classes	34
188	Table 55 – Literals of	
189	RemedialActionScheduleProfile::RemedialActionScheduleDependencyKind	34
190		

191 1 Introduction

192 The remedial action schedule profile is a profile to exchange a list of proposed, agreed, rejected,
193 etc. remedial action schedules.

194 Each grid state alteration defined as part of an available remedial action (by the remedial action
195 profile) gets a schedule for the parameter that should be modified when the remedial action
196 schedule is agreed and ordered. The remedial action schedule profile allows for several data
197 exchanges:

- 198 - List of remedial action schedules as output from a security analysis
- 199 - An exchange of the status of the remedial action
- 200 - An exchange of the agreements per TSO.

201

202 2 Application profile specification

203 2.1 Version information

204 The content is generated from UML model file CIM17-2_CGMES31v01_PROF-
205 20v02_NC23v66_MS10v01_DES10v01.eap.

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

- 207 - Title: Remedial Action Schedule Vocabulary
- 208 - Keyword: RAS
- 209 - Description: This vocabulary is describing the remedial action schedule profile.
- 210 - Version IRI: <https://ap-voc.cim4.eu/RemedialActionSchedule/2.3>
- 211 - Version info: 2.3.1
- 212 - Prior version: <http://entsoe.eu/ns/CIM/RemedialActionSchedule-EU/2.2>
- 213 - Conforms to: urn:iso:std:iec:61970-600-2:ed-1|urn:iso:std:iec:61970-301:ed-
214 7:amd1|file://iec61970cim17v40_iec61968cim13v13a_iec62325cim03v17a.eap|urn:iso:
215 std:iec:61970-401:draft:ed-1|urn:iso:std:iec:61970-501:draft:ed-
216 2|file://CIM100_CGMES31v01_501-20v02_NC23v62_MM10v01.eap
- 217 - Identifier: urn:uuid:6e90c546-3c6c-471b-8040-e05037081c59

218

219 2.2 Constraints naming convention

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

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

223 where

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

225 rule.Standard: the number of the standard e.g. 301 for 61970-301, 456 for 61970-456, 13 for
226 61968-13. 61970-600 specific constraints refer to 600 although they are related to one or
227 combination of the 61970-450 series profiles. For NC profiles, NC is used.

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

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

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

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

235 2.3 Profile constraints

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

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

- 241 • C:452:ALL:NA:datatypes

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

- 244 • R:452:ALL:NA:exchange

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

- 247 • R:452:ALL:NA:exchange1

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

- 253 • R:452:ALL:NA:exchange2

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

- 262 • R:452:ALL:NA:exchange3

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

- 266 • R:452:ALL:NA:exchange4

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

- 278 • R:452:ALL:NA:cardinality

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

- 284 • R:452:ALL:NA:associations

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

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

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

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

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

- 299 • R:452:ALL:NA:uniqueIdentifier

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

- 302 • R:452:ALL:NA:unitMultiplier

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

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

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

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

308 The string IdentifiedObject.description is maximum 256 characters.

309 • C:452:ALL:NA:float

310 An attribute that is defined as float (e.g. has a type Float or a type which is a Datatype
311 with .value attribute of type Float) shall support ISO/IEC 60559:2020 for floating-point
312 arithmetic using single precision floating point. A single precision float supports 7
313 significant digits where the significant digits are described as an integer, or a decimal
314 number with 6 decimal digits. Two float values are equal when the significant with 7
315 digits are identical, e.g. 1234567 is equal 1.234567E6 and so are 1.2345678 and
316 1.234567E0.

317 • R:NC:ALL:Region:reference

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

320 • R:NC:ALL:SystemOperator:reference

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

322 • C:NC:RAS:RemedialActionSchedule:proposingEntity

323 The RemedialActionSchedule shall have a proposing entity either the Security
324 Coordinator (RemedialActionSchedule.ProposingEntity) or at least one
325 ProposingRemedialActionScheduleShare.

326 • R:NC:ALL:NA:serialization

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

- 329 1) the EAP file (EnterpriseArchitect project file),
- 330 2) the profiles' specification document and
- 331 3) the application profiles (RDFS and SHACL).

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

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

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

350 It should be taken into account that this approach deviates from MVAL5 (IEC 61970-
351 600-1:2021), which creates multiple inheritance at serialization. For instance, with this
352 more explicit exchange the serialization of the association between abstract class
353 Equipment and abstract class Circuit for a PowerTransformer will be serialized as
354 follows:

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

359 ○ for attribute
360 <cim:ACLineSegment rdf:about="_04f681aa-6999-4fb3-9775-aca5eb7ceff">
361 <cim:Equipment.inService>true</cim:Equipment.inService>
362 </cim:ACLineSegment>

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

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

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

369

370 2.4 Metadata

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

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

380 2.4.1 Constraints

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

- 383 • R:NC:ALL:wasAttributedTo:usage

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

386

387 2.4.2 Reference metadata

388 The header defined for this profile requires availability of a set of reference metadata. For
389 instance, the attribute prov:wasGeneratedBy requires a reference to an activity which produced
390 the model or the related process. The activities are defined as reference metadata and their
391 identifiers are referenced from the header to enable the receiving entity to retrieve the "static"
392 (reference) information that is not modified frequently. This approach imposes a requirement
393 that both the sending entity and the receiving entity have access to a unique version of the

413 **Table 2 – Attributes of RemedialActionScheduleProfile::BaseTimeSeries**

name	mult	type	description
interpolationKind	1..1	TimeSeriesInterpolationKind	(NC) Kind of interpolation done between time point.
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

414

415 **3.4 (abstract) Contingency root class**

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

417 **3.5 (NC) EventSchedule**418 Inheritance path = [BaseIrregularTimeSeries](#) : [BaseTimeSeries](#) : [IdentifiedObject](#)

419 Time series represent irregular event described by event points in time.

420 Table 3 shows all attributes of EventSchedule.

421 **Table 3 – Attributes of RemedialActionScheduleProfile::EventSchedule**

name	mult	type	description
interpolationKind	1..1	TimeSeriesInterpolationKind	(NC) inherited from: BaseTimeSeries
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

422

423 Table 4 shows all association ends of EventSchedule with other classes.

424 **Table 4 – Association ends of RemedialActionScheduleProfile::EventSchedule with other classes**

425

mult from	name	mult to	type	description
0..1	RemedialActionSchedule	0..1	RemedialActionSchedule	Remedial action schedule is the event that is validity for the given time series.

426

427 **3.6 (NC) EventTimePoint root class**

428 Event valid for a given point in time.

429 Table 5 shows all attributes of EventTimePoint.

430 **Table 5 – Attributes of RemedialActionScheduleProfile::EventTimePoint**

name	mult	type	description
atTime	0..1	DateTime	(NC) The time the data is valid for.
isActive	0..1	Boolean	(NC) True, if the event is occurring (Active) at this time point. Otherwise false.

431

432 Table 6 shows all association ends of EventTimePoint with other classes.

433 **Table 6 – Association ends of RemedialActionScheduleProfile::EventTimePoint with**
434 **other classes**

mult from	name	mult to	type	description
1..*	EventSchedule	1..1	EventSchedule	(NC) Time series the time point values belongs to.

435

436 3.7 (abstract,NC) GenericValueSchedule

437 Inheritance path = [BaseIrregularTimeSeries](#) : [BaseTimeSeries](#) : [IdentifiedObject](#)

438 Time series represent irregular generic value at given points in time. The type of value is given
439 by the reference association.

440 Table 7 shows all attributes of GenericValueSchedule.

441 **Table 7 – Attributes of RemedialActionScheduleProfile::GenericValueSchedule**

name	mult	type	description
interpolationKind	1..1	TimeSeriesInterpolationKind	(NC) inherited from: BaseTimeSeries
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

442

443 Table 8 shows all association ends of GenericValueSchedule with other classes.

444 **Table 8 – Association ends of RemedialActionScheduleProfile::GenericValueSchedule**
445 **with other classes**

mult from	name	mult to	type	description
0..*	RemedialActionSchedule	0..1	RemedialActionSchedule	(NC) Remedial action schedule which has generic value schedules.

446

447 3.8 (NC) GenericValueTimePoint root class

448 Generic value for a given point in time.

449 Table 9 shows all attributes of GenericValueTimePoint.

450 **Table 9 – Attributes of RemedialActionScheduleProfile::GenericValueTimePoint**

name	mult	type	description
atTime	1..1	DateTime	(NC) The time the data is valid for.
value	1..1	Float	(NC) The value at the time. The meaning of the value is defined by the derived type of the associated schedule. The value can be integer, float or boolean. In case of boolean 1 equals true and 0 equals false.

451

452 Table 10 shows all association ends of GenericValueTimePoint with other classes.

453 **Table 10 – Association ends of RemedialActionScheduleProfile::GenericValueTimePoint**
454 **with other classes**

mult from	name	mult to	type	description
1..*	GenericValueSchedule	1..1	GenericValueSchedule	(NC) Time series the time point values belongs to.

455

456 3.9 (abstract,NC) GridStateAlteration

457 Inheritance path = [IdentifiedObject](#)

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

460 Table 11 shows all attributes of GridStateAlteration.

461 **Table 11 – Attributes of RemedialActionScheduleProfile::GridStateAlteration**

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

462

463 3.10 (NC) GridStateIntensitySchedule

464 Inheritance path = [GenericValueSchedule](#) : [BaseIrregularTimeSeries](#) : [BaseTimeSeries](#) :
465 [IdentifiedObject](#)

466 Defines the intensity applied for a given grid state alteration. It is primarily used in exchanges
467 related to the remedial action schedule. The value provided by the schedule replaces the value
468 of the attribute to which the schedule refers to.

469 Table 12 shows all attributes of GridStateIntensitySchedule.

470 **Table 12 – Attributes of RemedialActionScheduleProfile::GridStateIntensitySchedule**

name	mult	type	description
valueKind	0..1	ValueOffsetKind	(NC) The kind of value1 and value2 of the associated IrregularIntervalSchedule.
interpolationKind	1..1	TimeSeriesInterpolationKind	(NC) inherited from: BaseTimeSeries
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

471

472 Table 13 shows all association ends of GridStateIntensitySchedule with other classes.

473 **Table 13 – Association ends of**
474 **RemedialActionScheduleProfile::GridStateIntensitySchedule with other classes**

mult from	name	mult to	type	description
0..*	GridStateAlteration	1..1	GridStateAlteration	(NC) The grid state alteration which has intensity.
0..*	RemedialActionSchedule	0..1	RemedialActionSchedule	(NC) inherited from: GenericValueSchedule

475

476 **3.11 (abstract) IdentifiedObject root class**477 This is a root class to provide common identification for all classes needing identification and
478 naming attributes.

479 Table 14 shows all attributes of IdentifiedObject.

480 **Table 14 – Attributes of RemedialActionScheduleProfile::IdentifiedObject**

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

481

482 **3.12 (abstract,NC) RemedialAction**483 Inheritance path = [IdentifiedObject](#)484 Remedial action describes one or more actions that can be performed on a given power system
485 model situation to eliminate one or more identified breaches of constraints. The remedial action
486 can be costly, and have a cost characteristic, or non costly.

487 Table 15 shows all attributes of RemedialAction.

488 **Table 15 – Attributes of RemedialActionScheduleProfile::RemedialAction**

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

489

490 **3.13 (NC) RemedialActionCost root class**

491 Remedial action cost is the total cost itemised cost by category and type for the remedial action.

492 Table 16 shows all attributes of RemedialActionCost.

493 **Table 16 – Attributes of RemedialActionScheduleProfile::RemedialActionCost**

name	mult	type	description
costAllocationTime	0..1	DateTime	(NC) Cost allocation time is the time the cost shall be allocated.
kind	1..1	CostSettledKind	(NC) Remedial action cost category related to the confirmation of the cost in regards to changes.
operationalCost	0..1	Decimal	(NC) Operational cost is the total cost directly related to operate the unit according to the remedial action, e.g. fuel cost.

name	mult	type	description
opportunityCost	0..1	Decimal	(NC) Opportunity cost is the total cost of potential earning that is missed due to performing the remedial action.
otherCost	0..1	Decimal	(NC) Other cost is the total cost that cannot be directly allocated to any of the other items.
processingFee	0..1	Decimal	(NC) Processing fee is the total cost for processing the remedial action.
savedFuelCost	0..1	Decimal	(NC) Saved fuel cost is the total saving due to not consuming the expected fuel as part of the remedial action.
shutdownCost	0..1	Decimal	(NC) Shutdown cost is the total cost for shutting down a unit as part of the remedial action.
startupCost	0..1	Decimal	(NC) Start-up cost is the total cost for activating the remedial action, e.g. if a generator needs to be started before it can perform the remedial action.
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.

494

495

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

496

Table 17 – Association ends of RemedialActionScheduleProfile::RemedialActionCost with other classes

497

mult from	name	mult to	type	description
0..*	RemedialActionSchedule	1..1	RemedialActionSchedule	(NC) Remedial action schedule for which this remedial action cost relates to.

498

3.14 (NC) RemedialActionSchedule

499

Inheritance path = [IdentifiedObject](#)

500

501

A schedule for a determined remedial action.

502

Table 18 shows all attributes of RemedialActionSchedule.

503

Table 18 – Attributes of RemedialActionScheduleProfile::RemedialActionSchedule

name	mult	type	description
statusKind	1..1	RemedialActionScheduleStatusKind	(NC) Indicates the status kind for the remedial action schedule.
statusReason	0..1	String	(NC) Description of reasoning for the status. For instance, in case of rejected remedial action, the reason for this rejection is described here.
totalCostCurrency	0..1	Currency	(NC) The currency of the total cost.
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

504

505 Table 19 shows all association ends of RemedialActionSchedule with other classes.

506

507

Table 19 – Association ends of RemedialActionScheduleProfile::RemedialActionSchedule with other classes

mult from	name	mult to	type	description
0..*	Contingency	0..1	Contingency	(NC) The contingency for a curative remedial action schedule.
0..*	RemedialAction	1..1	RemedialAction	(NC) The remedial action that has a remedial action schedule associated.
0..*	ProposingEntity	0..1	SecurityCoordinator	(NC) The security coordinator that is proposing this remedial action schedule.
0..*	AssignedRegion	0..1	Region	(NC) The assigned region for this remedial action schedule.

508

509 3.15 (NC) RemedialActionScheduleAcceptance root class

510 It identifies if the remedial action schedule is accepted for a given system operator.

511 Table 20 shows all attributes of RemedialActionScheduleAcceptance.

512

513

Table 20 – Attributes of RemedialActionScheduleProfile::RemedialActionScheduleAcceptance

name	mult	type	description
kind	1..1	RemedialActionScheduleAcceptanceKind	(NC) The kind of the remedial action acceptance.
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.

514

515 Table 21 shows all association ends of RemedialActionScheduleAcceptance with other classes.

516

517

518

Table 21 – Association ends of RemedialActionScheduleProfile::RemedialActionScheduleAcceptance with other classes

mult from	name	mult to	type	description
0..*	RemedialActionSchedule	1..1	RemedialActionSchedule	(NC) A remedial action schedule for which a remedial action schedule acceptance is reported.
0..*	RespondingEntity	1..1	SystemOperator	(NC) An entity for which a remedial action schedule acceptances are reported.

519

520 3.16 (abstract,NC) Region root class

521 A region where the system operator belongs to.

522 **3.17 (abstract,NC) ScheduleResource root class**

523 A schedule resource is a market-based method for handling participation of small units,
524 particularly located on the lower voltage level that is controlled by a Distributed System
525 Operator (DSO). It is a collection of units that can operate in the market by providing bids, offers
526 and a resulting committed operational schedule for the collection.

527 **3.18 (abstract,NC) SecurityCoordinator root class**

528 A role that coordinates the relevant remedial actions and their optimisation to ensure efficient
529 use to achieve required operational security of the power system.

530 **3.19 (abstract,NC) SystemOperator root class**

531 System operator.

532 **3.20 (NC) CostSettledKind enumeration**

533 Kind describing how settled the cost is in regards to changes.

534 Table 22 shows all literals of CostSettledKind.

535 **Table 22 – Literals of RemedialActionScheduleProfile::CostSettledKind**

literal	value	description
indicative		Indicative cost.
provisional		Provisional cost.
final		Final cost. For instance, the cost is not expected to be changed on a later stage.

536

537 **3.21 (NC) RemedialActionScheduleAcceptanceKind enumeration**

538 The kind of acceptance for a remedial action schedule.

539 Table 23 shows all literals of RemedialActionScheduleAcceptanceKind.

540 **Table 23 – Literals of RemedialActionScheduleProfile::RemedialActionScheduleAcceptanceKind**

literal	value	description
accepted		The acceptance of remedial action schedule is concluded and accepted.
refused		The acceptance of the remedial action schedule is concluded and refused.
waiting		The acceptance of the remedial action schedule is waiting (in progress).
timeout		The acceptance of the remedial action schedule was not completed due to timeout.

542

543 **3.22 (NC) TimeSeriesInterpolationKind enumeration**

544 Kinds of interpolation of values between two time point.

545 Table 24 shows all literals of TimeSeriesInterpolationKind.

546 **Table 24 – Literals of RemedialActionScheduleProfile::TimeSeriesInterpolationKind**

literal	value	description
none		No interpolation is applied.
next		The value between two time points is set to next value.

547

548 **3.23 (NC) RemedialActionScheduleStatusKind enumeration**

549 Remedial action schedule status kinds.

550 Table 25 shows all literals of RemedialActionScheduleStatusKind.

551

552

**Table 25 – Literals of
RemedialActionScheduleProfile::RemedialActionScheduleStatusKind**

literal	value	description
proposed		Proposed remedial action schedule.
agreed		Agreed remedial action schedule.
rejected		Rejected remedial action schedule.
ordered		Ordered remedial action schedule.
previouslyAgreed		Previously agreed remedial action schedule.
notUsed		Not used remedial action schedule.
agreementValidated		The agreement is validated for the remedial action schedule.
rejectionValidated		The rejection is validated for the remedial action schedule.
implemented		An ordered remedial action is implemented.
activated		Activated remedial action schedule.

553

554 **3.24 UnitMultiplier enumeration**

555 The unit multipliers defined for the CIM. When applied to unit symbols, the unit symbol is
556 treated as a derived unit. Regardless of the contents of the unit symbol text, the unit symbol
557 shall be treated as if it were a single-character unit symbol. Unit symbols should not contain
558 multipliers, and it should be left to the multiplier to define the multiple for an entire data type.

559 For example, if a unit symbol is "m2Pers" and the multiplier is "k", then the value is $k(m^{**2}/s)$,
560 and the multiplier applies to the entire final value, not to any individual part of the value. This
561 can be conceptualized by substituting a derived unit symbol for the unit type. If one imagines
562 that the symbol "P" represents the derived unit "m2Pers", then applying the multiplier "k" can
563 be conceptualized simply as "kP".

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

572 Table 26 shows all literals of UnitMultiplier.

573

Table 26 – Literals of RemedialActionScheduleProfile::UnitMultiplier

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

574

575 **3.25 UnitSymbol enumeration**

576 The derived units defined for usage in the CIM. In some cases, the derived unit is equal to an
577 SI unit. Whenever possible, the standard derived symbol is used instead of the formula for the
578 derived unit. For example, the unit symbol Farad is defined as "F" instead of "CPerV". In cases
579 where a standard symbol does not exist for a derived unit, the formula for the unit is used as

580 the unit symbol. For example, density does not have a standard symbol and so it is represented
581 as "kgPerm3". With the exception of the "kg", which is an SI unit, the unit symbols do not contain
582 multipliers and therefore represent the base derived unit to which a multiplier can be applied as
583 a whole.

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

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

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

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

598 Table 27 shows all literals of UnitSymbol.

599

Table 27 – Literals of RemedialActionScheduleProfile::UnitSymbol

literal	value	description
s	4	Time in seconds.
W	38	Real power in watts (J/s). Electrical power may have real and reactive components. The real portion of electrical power (I^2R or $V\cos(\phi)$), is expressed in Watts. See also apparent power and reactive power.
VAr	63	Reactive power in volt amperes reactive. The "reactive" or "imaginary" component of electrical power ($V\sin(\phi)$). (See also real power and apparent power). Note: Different meter designs use different methods to arrive at their results. Some meters may compute reactive power as an arithmetic value, while others compute the value vectorially. The data consumer should determine the method in use and the suitability of the measurement for the intended purpose.
Wh	72	Real energy in watt hours.

600

601 3.26 (NC) ValueOffsetKind enumeration

602 The kind of the value offset.

603 Table 28 shows all literals of ValueOffsetKind.

604

Table 28 – Literals of RemedialActionScheduleProfile::ValueOffsetKind

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

literal	value	description
		PropertyReference in a determined operational scenario.

605

606 3.27 ActivePower datatype

607 Product of RMS value of the voltage and the RMS value of the in-phase component of the
608 current.

609 Table 29 shows all attributes of ActivePower.

610 **Table 29 – Attributes of RemedialActionScheduleProfile::ActivePower**

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

611

612 3.28 RealEnergy datatype

613 Real electrical energy.

614 Table 30 shows all attributes of RealEnergy.

615 **Table 30 – Attributes of RemedialActionScheduleProfile::RealEnergy**

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

616

617 3.29 Seconds datatype

618 Time, in seconds.

619 Table 31 shows all attributes of Seconds.

620 **Table 31 – Attributes of RemedialActionScheduleProfile::Seconds**

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

621

622 3.30 Boolean primitive

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

624 3.31 DateTime primitive

625 Date and time as "yyyy-mm-ddThh:mm:ss.sss", which conforms with ISO 8601. UTC time zone
626 is specified as "yyyy-mm-ddThh:mm:ss.sssZ". A local timezone relative UTC is specified as
627 "yyyy-mm-ddThh:mm:ss.sss-hh:mm". The second component (shown here as "ss.sss") could
628 have any number of digits in its fractional part to allow any kind of precision beyond seconds.

629 3.32 Duration primitive

630 Duration as "PnYnMnDnHnMnS" which conforms to ISO 8601, where nY expresses a number
631 of years, nM a number of months, nD a number of days. The letter T separates the date
632 expression from the time expression and, after it, nH identifies a number of hours, nM a number

633 of minutes and nS a number of seconds. The number of seconds could be expressed as a
634 decimal number, but all other numbers are integers.

635 3.33 Decimal primitive

636 Decimal is the base-10 notational system for representing real numbers.

637 3.34 Float primitive

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

639 3.35 String primitive

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

642 3.36 (NC,Description) RemedialActionSchemeSchedule

643 Inheritance path = [BaseIrregularTimeSeries](#) : [BaseTimeSeries](#) : [IdentifiedObject](#)

644 The schedule for a remedial action scheme.

645 Table 32 shows all attributes of RemedialActionSchemeSchedule.

646
647

Table 32 – Attributes of RemedialActionScheduleProfile::RemedialActionSchemeSchedule

name	mult	type	description
interpolationKind	1..1	TimeSeriesInterpolationKind	(NC) inherited from: BaseTimeSeries
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

648
649

Table 33 shows all association ends of RemedialActionSchemeSchedule with other classes.

650
651

Table 33 – Association ends of RemedialActionScheduleProfile::RemedialActionSchemeSchedule with other classes

mult from	name	mult to	type	description
0..1	SchemeRemedialAction	0..1	SchemeRemedialAction	(NC) Remedial action scheme which has remedial action scheme schedules.

652

653 3.37 (NC) PowerSchedule

654 Inheritance path = [BaseIrregularTimeSeries](#) : [BaseTimeSeries](#) : [IdentifiedObject](#)

655 Time series represent irregular power, active and reactive, values at given points in time.

656 Table 34 shows all attributes of PowerSchedule.

657

Table 34 – Attributes of RemedialActionScheduleProfile::PowerSchedule

name	mult	type	description
interpolationKind	1..1	TimeSeriesInterpolationKind	(NC) inherited from: BaseTimeSeries
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

658

659 3.38 (NC) PowerTimePoint root class

660 Power, active and reactive, value at a given point in time.

661 Table 35 shows all attributes of PowerTimePoint.

662

Table 35 – Attributes of RemedialActionScheduleProfile::PowerTimePoint

name	mult	type	description
atTime	1..1	DateTime	(NC) The time the data is valid for.
p	0..1	ActivePower	(NC) Active power injection. Load sign convention is used, i.e. positive sign means flow out from a node.
q	0..1	ReactivePower	(NC) Reactive power injection. Load sign convention is used, i.e. positive sign means flow out from a node.
price	0..1	Decimal	(NC) Price for the scheduled active power per unit of active power. e.g. per MW.
activatedP	0..1	ActivePower	(NC) Active power activated as part of redispatch. Negative number means that the value is scheduling down. Positive number means that the value is scheduling up.
activatedPrice	0..1	Decimal	(NC) Price for the activated active power per unit e.g. per MW.

663

664

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

665

666

Table 36 – Association ends of RemedialActionScheduleProfile::PowerTimePoint with other classes

mult from	name	mult to	type	description
0..*	PowerSchedule	1..1	PowerSchedule	(NC) Time series the time point values belongs to.

667

668 3.39 ReactivePower datatype

669 Product of RMS value of the voltage and the RMS value of the quadrature component of the
670 current.

671 Table 37 shows all attributes of ReactivePower.

672

Table 37 – Attributes of RemedialActionScheduleProfile::ReactivePower

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

673

674 3.40 Integer primitive

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

676

677 3.41 (abstract,NC) PowerBidSchedule

678 Inheritance path = [BaseIrregularTimeSeries](#) : [BaseTimeSeries](#) : [IdentifiedObject](#)

679 Power bid or offer related to a redispatch or countertrading measures. In the case of market
680 place for economic efficiency of the bids and offers, this is equivalent to BidTimeSeries class
681 in 62325 package.

Table 38 shows all attributes of PowerBidSchedule.

682 **Table 38 – Attributes of RemedialActionScheduleProfile::PowerBidSchedule**

name	mult	type	description
interpolationKind	1..1	TimeSeriesInterpolationKind	(NC) inherited from: BaseTimeSeries
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

683

684 **3.42 (abstract,NC) AvailabilityRemedialAction**685 Inheritance path = [RemedialAction](#) : [IdentifiedObject](#)686 Availability remedial action is a remedial action that cancels or reschedules an availability
687 schedule.

688 Table 39 shows all attributes of AvailabilityRemedialAction.

689 **Table 39 – Attributes of RemedialActionScheduleProfile::AvailabilityRemedialAction**

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

690

691 **3.43 (NC,Description) SchemeRemedialAction**692 Inheritance path = [RemedialAction](#) : [IdentifiedObject](#)693 Scheme remedial action is remedial action that involves a scheme that can include conditional
694 logic and stages of grid alteration. The primary remedial action is the arming of these schemes,
695 that will then perform curative remedial action when the condition is met. System Integrity
696 Protection Scheme (SIPS) and Special Protection Scheme (SPS) are example of this.

697 Table 40 shows all attributes of SchemeRemedialAction.

698 **Table 40 – Attributes of RemedialActionScheduleProfile::SchemeRemedialAction**

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

699

700 **3.44 (NC,Description) AvailabilitySchedule**701 Inheritance path = [IdentifiedObject](#)702 A given (un)availability schedule with a given status and cause that include multiple equipment
703 that need to follow the same scheduling periods.

704 Table 41 shows all attributes of AvailabilitySchedule.

705 **Table 41 – Attributes of RemedialActionScheduleProfile::AvailabilitySchedule**

name	mult	type	description
isCancelled	1..1	Boolean	(NC) Defines the cancelling of the availability schedule. True means that is cancelling, False means that it is not cancelling.
priority	1..1	Integer	(NC) Value 0 means ignore priority. 1 means the highest priority, 2 is the second highest priority.

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

706

707

Table 42 shows all association ends of AvailabilitySchedule with other classes.

708

709

Table 42 – Association ends of RemedialActionScheduleProfile::AvailabilitySchedule with other classes

mult from	name	mult to	type	description
1..*	RemedialAction	1..1	AvailabilityRemedialAction	(NC) Remedial action that is cancelling this availability schedule.

710

711

3.45 Currency enumeration

712

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

713

Table 43 shows all literals of Currency.

714

Table 43 – Literals of RemedialActionScheduleProfile::Currency

literal	value	description
AED	784	United Arab Emirates dirham.
AFN	971	Afghan afghani.
ALL	008	Albanian lek.
AMD	051	Armenian dram.
ANG	532	Netherlands Antillean guilder.
AOA	973	Angolan kwanza.
ARS	032	Argentine peso.
AUD	036	Australian dollar.
AWG	533	Aruban florin.
AZN	944	Azerbaijani manat.
BAM	977	Bosnia and Herzegovina convertible mark.
BBD	052	Barbados dollar.
BDT	050	Bangladeshi taka.
BGN	975	Bulgarian lev.
BHD	048	Bahraini dinar.
BIF	108	Burundian franc.
BMD	060	Bermudian dollar (customarily known as Bermuda dollar).
BND	096	Brunei dollar.
BOB	068	Boliviano.
BOV	984	Bolivian Mvdol (funds code).
BRL	986	Brazilian real.
BSD	044	Bahamian dollar.
BTN	064	Bhutanese ngultrum.
BWP	072	Botswana pula.

literal	value	description
BYR	974	Belarusian ruble.
BZD	084	Belize dollar.
CAD	124	Canadian dollar.
CDF	976	Congolese franc.
CHF	756	Swiss franc.
CLF	990	Unidad de Fomento (funds code), Chile.
CLP	152	Chilean peso.
CNY	156	Chinese yuan.
COP	170	Colombian peso.
COU	970	Unidad de Valor Real.
CRC	188	Costa Rican colon.
CUC	931	Cuban convertible peso.
CUP	192	Cuban peso.
CVE	132	Cape Verde escudo.
CZK	203	Czech koruna.
DJF	262	Djiboutian franc.
DKK	208	Danish krone.
DOP	214	Dominican peso.
DZD	012	Algerian dinar.
EEK	233	Estonian kroon.
EGP	818	Egyptian pound.
ERN	232	Eritrean nakfa.
ETB	230	Ethiopian birr.
EUR	978	Euro.
FJD	242	Fiji dollar.
FKP	238	Falkland Islands pound.
GBP	826	Pound sterling.
GEL	981	Georgian lari.
GHS	936	Ghanaian cedi.
GIP	929	Gibraltar pound.
GMD	270	Gambian dalasi.
GNF	324	Guinean franc.
GTQ	320	Guatemalan quetzal.
GYD	328	Guyanese dollar.
HKD	344	Hong Kong dollar.
HNL	340	Honduran lempira.
HRK	191	Croatian kuna.
HTG	332	Haitian gourde.
HUF	348	Hungarian forint.
IDR	360	Indonesian rupiah.
ILS	376	Israeli new sheqel.

literal	value	description
INR	356	Indian rupee.
IQD	368	Iraqi dinar.
IRR	364	Iranian rial.
ISK	352	Icelandic króna.
JMD	388	Jamaican dollar.
JOD	400	Jordanian dinar.
JPY	392	Japanese yen.
KES	404	Kenyan shilling.
KGS	417	Kyrgyzstani som.
KHR	116	Cambodian riel.
KMF	174	Comoro franc.
KPW	408	North Korean won.
KRW	410	South Korean won.
KWD	414	Kuwaiti dinar.
KYD	136	Cayman Islands dollar.
KZT	398	Kazakhstani tenge.
LAK	418	Lao kip.
LBP	422	Lebanese pound.
LKR	144	Sri Lanka rupee.
LRD	430	Liberian dollar.
LSL	426	Lesotho loti.
LTL	440	Lithuanian litas.
LVL	428	Latvian lats.
LYD	434	Libyan dinar.
MAD	504	Moroccan dirham.
MDL	498	Moldovan leu.
MGA	969	Malagasy ariary.
MKD	807	Macedonian denar.
MMK	104	Myanma kyat.
MNT	496	Mongolian tugrik.
MOP	446	Macanese pataca.
MRO	478	Mauritanian ouguiya.
MUR	480	Mauritian rupee.
MVR	462	Maldivian rufiyaa.
MWK	454	Malawian kwacha.
MXN	484	Mexican peso.
MYR	458	Malaysian ringgit.
MZN	943	Mozambican metical.
NAD	516	Namibian dollar.
NGN	566	Nigerian naira.
NIO	558	Cordoba oro.

literal	value	description
NOK	578	Norwegian krone.
NPR	524	Nepalese rupee.
NZD	554	New Zealand dollar.
OMR	512	Omani rial.
PAB	590	Panamanian balboa.
PEN	604	Peruvian nuevo sol.
PGK	598	Papua New Guinean kina.
PHP	608	Philippine peso.
PKR	586	Pakistani rupee.
PLN	985	Polish zloty.
PYG	600	Paraguayan guaraní.
QAR	634	Qatari rial.
RON	946	Romanian new leu.
RSD	941	Serbian dinar.
RUB	643	Russian rouble.
RWF	646	Rwandan franc.
SAR	682	Saudi riyal.
SBD	090	Solomon Islands dollar.
SCR	690	Seychelles rupee.
SDG	938	Sudanese pound.
SEK	752	Swedish krona/kronor.
SGD	702	Singapore dollar.
SHP	654	Saint Helena pound.
SLL	694	Sierra Leonean leone.
SOS	706	Somali shilling.
SRD	968	Surinamese dollar.
STD	678	São Tomé and Príncipe dobra.
SYP	760	Syrian pound.
SZL	748	Lilangeni.
THB	764	Thai baht.
TJS	972	Tajikistani somoni.
TMT	934	Turkmenistani manat.
TND	788	Tunisian dinar.
TOP	776	Tongan pa'anga.
TRY	949	Turkish lira.
TTD	780	Trinidad and Tobago dollar.
TWD	901	New Taiwan dollar.
TZS	834	Tanzanian shilling.
UAH	980	Ukrainian hryvnia.
UGX	800	Ugandan shilling.
USD	840	United States dollar.

literal	value	description
UYU	858	Uruguayan peso.
UZS	860	Uzbekistan som.
VEF	937	Venezuelan bolívar fuerte.
VND	704	Vietnamese Dong.
VUV	548	Vanuatu vatu.
WST	882	Samoan tala.
XAF	950	CFA franc BEAC.
XCD	951	East Caribbean dollar.
XOF	952	CFA Franc BCEAO.
XPF	953	CFP franc.
YER	886	Yemeni rial.
ZAR	710	South African rand.
ZMK	894	Zambian kwacha.
ZWL	932	Zimbabwe dollar.

715

716 **3.46 (NC) ProposingRemedialActionScheduleShare root class**717 Proposing entity (System Operator) with a proper cost share for a given remedial action
718 schedule.

719 Table 44 shows all attributes of ProposingRemedialActionScheduleShare.

720

721

**Table 44 – Attributes of
RemedialActionScheduleProfile::ProposingRemedialActionScheduleShare**

name	mult	type	description
costSharingFactor	0..1	Decimal	(NC) Sharing factor of the cost of the remedial action as a fraction of the total cost, i.e. system operator's cost = cost x (costSharingFactor / sum of all costSharingFactor).
mRID	0..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.

722

723 Table 45 shows all association ends of ProposingRemedialActionScheduleShare with other
724 classes.

725

726

727

**Table 45 – Association ends of
RemedialActionScheduleProfile::ProposingRemedialActionScheduleShare with other
classes**

mult from	name	mult to	type	description
0..*	RemedialActionSchedule	1..1	RemedialActionSchedule	(NC) Remedial action schedule proposed by the proposing entity.

mult from	name	mult to	type	description
0..*	ProposingEntity	1..1	SystemOperator	(NC) Proposing entity making the proposing remedial action schedule share.

728

729 **3.47 (NC) RemedialActionScheduleGroup**730 Inheritance path = [IdentifiedObject](#)731 Remedial action schedule group collects two or more remedial action schedules together. The
732 remedial action schedule group needs to be set up for the same remedial action or proposing
733 alternative remedial action by including a reference to another remedial action.

734 Table 46 shows all attributes of RemedialActionScheduleGroup.

735

736

**Table 46 – Attributes of
RemedialActionScheduleProfile::RemedialActionScheduleGroup**

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

737

738 **3.48 (abstract,NC) PowerScheduleAction**739 Inheritance path = [IdentifiedObject](#)

740 Power schedule action is an action to rearrange power schedules.

741 Table 47 shows all attributes of PowerScheduleAction.

742

Table 47 – Attributes of RemedialActionScheduleProfile::PowerScheduleAction

name	mult	type	description
currency	0..1	Currency	(NC) Currency the energy price is given in.
energyPrice	0..1	Decimal	(NC) Energy price for the power schedule action.
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

743

744 Table 48 shows all association ends of PowerScheduleAction with other classes.

**Table 48 – Association ends of RemedialActionScheduleProfile::PowerScheduleAction
with other classes**

745

746

mult from	name	mult to	type	description
0..*	RemedialActionSchedule	1..1	RemedialActionSchedule	(NC) Remedial action schedule which power schedule actions.
0..1	PowerBidSchedule	0..1	PowerBidSchedule	(NC) The power bid schedule which contains the power schedule action.
0..1	PowerSchedule	1..1	PowerSchedule	(NC) Power schedule which contains the power schedule action.

747

748 **3.49 (NC) CountertradeScheduleAction**749 Inheritance path = [PowerScheduleAction](#) : [IdentifiedObject](#)

750 Countertrade schedule action is an action to rearrange power schedules based on a Generation
751 and Load Shift Key (GLSK) strategy.
752 Table 49 shows all attributes of CountertradeScheduleAction.

753 **Table 49 – Attributes of RemedialActionScheduleProfile::CountertradeScheduleAction**

name	mult	type	description
currency	0..1	Currency	(NC) inherited from: PowerScheduleAction
energyPrice	0..1	Decimal	(NC) inherited from: PowerScheduleAction
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

754
755 Table 50 shows all association ends of CountertradeScheduleAction with other classes.

756 **Table 50 – Association ends of**
757 **RemedialActionScheduleProfile::CountertradeScheduleAction with other classes**

mult from	name	mult to	type	description
0..*	RemedialActionSchedule	1..1	RemedialActionSchedule	(NC) inherited from: PowerScheduleAction
0..1	PowerBidSchedule	0..1	PowerBidSchedule	(NC) inherited from: PowerScheduleAction
0..1	PowerSchedule	1..1	PowerSchedule	(NC) inherited from: PowerScheduleAction

758

759 3.50 (NC) RedispatchScheduleAction

760 Inheritance path = [PowerScheduleAction](#) : [IdentifiedObject](#)
761 Redispatch schedule action is an action to rearrange power schedules for a scheduled resource
762 to obtain a feasible and secure operational state of the power electricity system.
763 Table 51 shows all attributes of RedispatchScheduleAction.

764 **Table 51 – Attributes of RemedialActionScheduleProfile::RedispatchScheduleAction**

name	mult	type	description
currency	0..1	Currency	(NC) inherited from: PowerScheduleAction
energyPrice	0..1	Decimal	(NC) inherited from: PowerScheduleAction
description	0..1	String	inherited from: IdentifiedObject
mRID	1..1	String	inherited from: IdentifiedObject
name	0..1	String	inherited from: IdentifiedObject

765

766 Table 52 shows all association ends of RedispatchScheduleAction with other classes.

767 **Table 52 – Association ends of**
768 **RemedialActionScheduleProfile::RedispatchScheduleAction with other classes**

mult from	name	mult to	type	description
0..*	ScheduleResource	0..1	ScheduleResource	(NC) The schedule resource that has this redispatch action.
0..*	RemedialActionSchedule	1..1	RemedialActionSchedule	(NC) inherited from: PowerScheduleAction

mult from	name	mult to	type	description
0..1	PowerBidSchedule	0..1	PowerBidSchedule	(NC) inherited from: PowerScheduleAction
0..1	PowerSchedule	1..1	PowerSchedule	(NC) inherited from: PowerScheduleAction

769

770 3.51 (NC) RemedialActionScheduleDependency root class

771 Remedial action schedule dependency is making two remedial action schedules depending on
772 each other.

773 Table 53 shows all attributes of RemedialActionScheduleDependency.

774

775

**Table 53 – Attributes of
RemedialActionScheduleProfile::RemedialActionScheduleDependency**

name	mult	type	description
kind	0..1	RemedialActionScheduleDependencyKind	(NC) Type of dependency between two remedial action schedules.
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.

776

777 Table 54 shows all association ends of RemedialActionScheduleDependency with other
778 classes.

779

780

781

**Table 54 – Association ends of
RemedialActionScheduleProfile::RemedialActionScheduleDependency with other
classes**

mult from	name	mult to	type	description
0..*	ReplacedRemedialActionSchedule	0..1	RemedialActionSchedule	(NC) Remedial action schedule that is replaced by dependent remedial action schedule.
0..*	ProposedRemedialActionSchedule	0..1	RemedialActionSchedule	(NC) Proposed remedial action schedule that has dependency.
0..*	RemedialActionScheduleGroup	0..1	RemedialActionScheduleGroup	(NC) Remedial action schedule group which has remedial action schedule dependency.

782

783 3.52 (NC) RemedialActionScheduleDependencyKind enumeration

784 Kind of dependency between remedial action schedules.

785 Table 55 shows all literals of RemedialActionScheduleDependencyKind.

786

787

**Table 55 – Literals of
RemedialActionScheduleProfile::RemedialActionScheduleDependencyKind**

literal	value	description
exclusive		Remedial action schedules are exclusive depending on each other. e.g. Only one of the

literal	value	description
		remedial action schedules can be selected at the same time.
inclusive		Remedial action schedules are inclusive depending on each other. e.g. Both remedial action schedules need to be picked if one of them is needed.
none		Remedial action schedules are not depending on each other. However, the two remedial action schedules should be evaluated together.

788

789

790

791

Annex A (informative): Sample data

792 **A.1 General**

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

797 **A.2 Sample instance data**

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

799