



European Network of
Transmission System Operators
for Electricity

IMPACT ASSESSMENT MATRIX PROFILE SPECIFICATION

2025-09-11

ICTC APPROVED
VERSION 2.4.0

Copyright notice:

Copyright © ENTSO-E. All Rights Reserved.

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

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

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

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

NOTE CONCERNING WORDING USED IN THIS DOCUMENT

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

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

Revision History

Version	Date	Paragraph	Comments
1.0.0	2021-03-22		Document for SOC approval
2.0.0	2021-10-12		For CIM EG review. No major update. Due to update of the extensions some elements are updated.
2.0.0	2022-02-16		No major update. Due to update of the extensions some elements are updated. 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	2024-03-20		For CIM WG review.
2.3.1-alpha	2024-09-07		For CIM WG review.
2.4.0	2025-09-11		ICTC approval.

CONTENTS

34			
35	Copyright notice:		2
36	Revision History		3
37	CONTENTS		4
38	1 Introduction		6
39	2 Application profile specification		6
40	2.1 Version information		6
41	2.2 Constraints naming convention		6
42	2.3 Profile constraints		7
43	2.4 Metadata		10
44	2.4.1 Constraints		11
45	2.4.2 Reference metadata		11
46	3 Detailed Profile Specification		11
47	3.1 General		11
48	3.2 (abstract,NC) RemedialActionGroup		12
49	3.3 (NC) CalculationBasedImpactAssessmentMatrix		12
50	3.4 (NC) ConnectingImpactAssessmentMatrix		13
51	3.5 (abstract) IdentifiedObject root class		13
52	3.6 (abstract,NC) ImpactAssessmentMatrix		13
53	3.7 (NC) ListBasedImpactAssessmentMatrix		14
54	3.8 (abstract,NC) OutcomeValue root class		14
55	3.9 (NC) OwnerRemedialActionAssessment root class		15
56	3.10 (NC) QualitativeRemedialActionImpact		15
57	3.11 (NC) QuantitativeRemedialActionImpact		16
58	3.12 (abstract,NC) RemedialAction root class		16
59	3.13 (abstract,NC) RemedialActionImpact root class		16
60	3.14 (NC) RemedialActionOutcomeValue		17
61	3.15 (abstract,NC) RemedialActionSchedule root class		18
62	3.16 (NC) RemedialActionScheduleOutcomeValue		18
63	3.17 (abstract,NC) SensitivityArea root class		18
64	3.18 (abstract,NC) SystemOperator root class		18
65	3.19 (NC) ImpactAgreementKind enumeration		18
66	3.20 (NC) OutcomeImpactAssessmentKind enumeration		19
67	3.21 Float primitive		19
68	3.22 String primitive		19
69	Annex A (informative): Sample data		20
70	A.1 General		20
71	A.2 Sample instance data		20
72			
73	List of figures		
74	Figure 1 – Class diagram		
75	ImpactAssessmentMatrixProfile::ImpactAssessmentMatrixProfile		12
76			

77 **List of tables**

78	Table 1 – Attributes of ImpactAssessmentMatrixProfile::RemedialActionGroup	12
79	Table 2 – Attributes of	
80	ImpactAssessmentMatrixProfile::CalculationBasedImpactAssessmentMatrix	13
81	Table 3 – Attributes of	
82	ImpactAssessmentMatrixProfile::ConnectingImpactAssessmentMatrix	13
83	Table 4 – Attributes of ImpactAssessmentMatrixProfile::IdentifiedObject	13
84	Table 5 – Attributes of ImpactAssessmentMatrixProfile::ImpactAssessmentMatrix	14
85	Table 6 – Attributes of	
86	ImpactAssessmentMatrixProfile::ListBasedImpactAssessmentMatrix	14
87	Table 7 – Attributes of ImpactAssessmentMatrixProfile::OutcomeValue	14
88	Table 8 – Association ends of ImpactAssessmentMatrixProfile::OutcomeValue with	
89	other classes	14
90	Table 9 – Attributes of	
91	ImpactAssessmentMatrixProfile::OwnerRemedialActionAssessment	15
92	Table 10 – Association ends of	
93	ImpactAssessmentMatrixProfile::OwnerRemedialActionAssessment with other classes	15
94	Table 11 – Attributes of	
95	ImpactAssessmentMatrixProfile::QualitativeRemedialActionImpact	15
96	Table 12 – Association ends of	
97	ImpactAssessmentMatrixProfile::QualitativeRemedialActionImpact with other classes	16
98	Table 13 – Attributes of	
99	ImpactAssessmentMatrixProfile::QuantitativeRemedialActionImpact	16
100	Table 14 – Association ends of	
101	ImpactAssessmentMatrixProfile::QuantitativeRemedialActionImpact with other classes	16
102	Table 15 – Attributes of ImpactAssessmentMatrixProfile::RemedialActionImpact	17
103	Table 16 – Association ends of ImpactAssessmentMatrixProfile::RemedialActionImpact	
104	with other classes	17
105	Table 17 – Attributes of	
106	ImpactAssessmentMatrixProfile::RemedialActionOutcomeValue	17
107	Table 18 – Association ends of	
108	ImpactAssessmentMatrixProfile::RemedialActionOutcomeValue with other classes	18
109	Table 19 – Attributes of	
110	ImpactAssessmentMatrixProfile::RemedialActionScheduleOutcomeValue	18
111	Table 20 – Association ends of	
112	ImpactAssessmentMatrixProfile::RemedialActionScheduleOutcomeValue with other	
113	classes	18
114	Table 21 – Literals of ImpactAssessmentMatrixProfile::ImpactAgreementKind	19
115	Table 22 – Literals of ImpactAssessmentMatrixProfile::OutcomeImpactAssessmentKind	19
116		

117 1 Introduction

118 The impact assessment matrix profile is a profile to exchange impact assessment matrices that
119 are needed within the process.

120 The impact assessment matrix is an output of the impact assessment done on proposed
121 remedial actions.

122 Three impact assessment matrices can be exchanged: list-based impact assessment matrix,
123 calculation-based impact assessment matrix and connecting impact assessment matrix.

124 2 Application profile specification

125 2.1 Version information

126 The content is generated from UML model file CIM17-2_CGMES31v01_PROF-
127 20v02_NC24v18_MS10v03_DES10v01.eap.

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

- 129 - Title: Impact Assessment Matrix Vocabulary
- 130 - Keyword: IAM
- 131 - Description: This vocabulary is describing the impact assessment matrix profile.
- 132 - Version IRI: <https://ap-voc.cim4.eu/ImpactAssessmentMatrix/2.4>
- 133 - Version info: 2.4.0
- 134 - Prior version: <https://ap-voc.cim4.eu/ImpactAssessmentMatrix/2.3>
- 135 - Conforms to: urn:iso:std:iec:61970-600-2:ed-1|urn:iso:std:iec:61970-301:ed-
136 7:amd1|file://iec61970cim17v40_iec61968cim13v13a_iec62325cim03v17a.eap|urn:iso:
137 std:iec:61970-401:draft:ed-1|urn:iso:std:iec:61970-501:draft:ed-2|file://CIM17-
138 2_CGMES31v01_PROF-20v02_NC24v18_MS10v03_DES10v01.eap
- 139 - Identifier: urn:uuid:1eb41c0b-3c58-4762-a79b-33220d051d32

140

141 2.2 Constraints naming convention

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

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

145 where

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

147 rule.Standard: the number of the standard e.g. 301 for 61970-301, 456 for 61970-456, 13 for
148 61968-13. 61970-600 specific constraints refer to 600 although they are related to one or
149 combination of the 61970-450 series profiles. For NC profiles, NC is used.

150 rule.Profile: the abbreviation of the profile, e.g. TP for Topology profile. If set to "ALL" the
151 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

236 number with 6 decimal digits. Two float values are equal when the significant with 7
237 digits are identical, e.g. 1234567 is equal 1.234567E6 and so are 1.2345678 and
238 1.234567E0.

239 • R:NC:ALL:Region:reference

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

242 • R:NC:ALL:SystemOperator:reference

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

244 • C:NC:IAM:ListBasedImpactAssessmentMatrix:outcomeValue

245 For a ListBasedImpactAssessmentMatrix, RemedialActionScheduleOutcomeValue shall
246 not be exchanged.

247 • C:NC:IAM:CalculationBasedImpactAssessmentMatrix:outcomeValue

248 For a CalculationBasedImpactAssessmentMatrix, RemedialActionOutcomeValue shall
249 not be exchanged.

250 • C:NC:IAM:ConnectingImpactAssessmentMatrix:outcomeValue

251 For a ConnectingImpactAssessmentMatrix, shall have either
252 RemedialActionScheduleOutcomeValue or RemedialActionOutcomeValue.

253 • R:NC:ALL:NA:serialization

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

- 256 1) the EAP file (EnterpriseArchitect project file),
- 257 2) the profiles' specification document and
- 258 3) the application profiles (RDFS and SHACL).

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

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

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

277 It should be taken into account that this approach deviates from MVAL5 (IEC 61970-
278 600-1:2021), which creates multiple inheritance at serialization. For instance, with this

279 more explicit exchange the serialization of the association between abstract class
280 Equipment and abstract class Circuit for a PowerTransformer will be serialized as
281 follows:

282 ○ for association

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

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

285 </cim:PowerTransformer>

286 ○ for attribute

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

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

289 </cim:ACLineSegment>

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

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

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

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

295 </cim:Equipment>

296 • C:NC:IAM:RemedialActionImpact:associations

297 A RemedialActionImpact shall have either RemedialActionGroup or RemedialAction
298 associated with it.

299 • C:NC:IAM:RemedialActionOutcomeValue:associations

300 A RemedialActionOutcomeValue shall have either RemedialActionGroup or
301 RemedialAction associated with it.

302

303

304

305 2.4 Metadata

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

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

316 2.4.1 Constraints

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

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

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

322

323 2.4.2 Reference metadata

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

332 3 Detailed Profile Specification

333 3.1 General

334 This package contains impact assessment matrix profile.

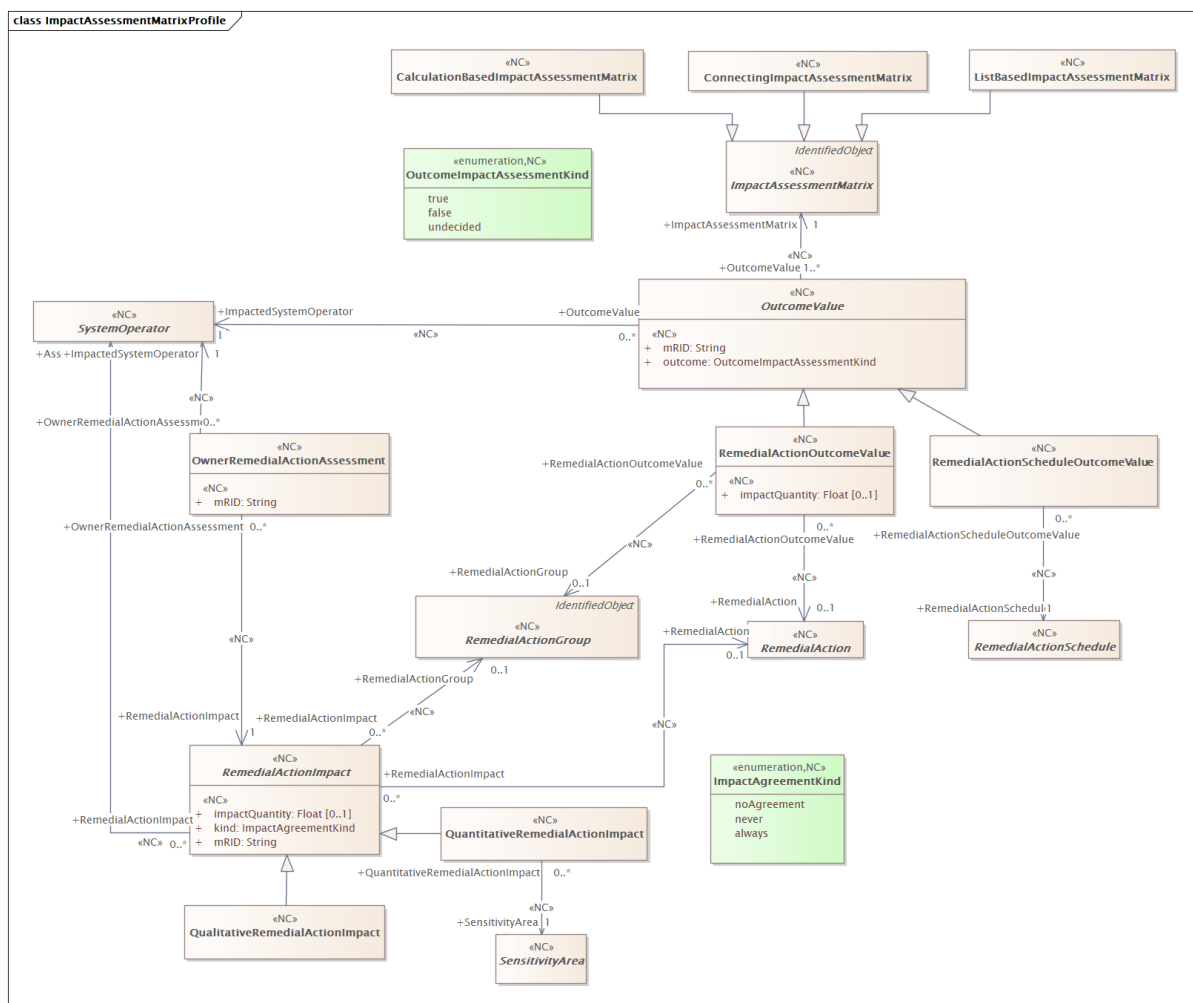


Figure 1 – Class diagram
ImpactAssessmentMatrixProfile::ImpactAssessmentMatrixProfile

Figure 1: The diagram contains the main classes used in the profile.

3.2 (abstract,NC) RemedialActionGroup

Inheritance path = IdentifiedObject

Grouping of remedial actions that can be operated together.

Table 1 shows all attributes of RemedialActionGroup.

Table 1 – Attributes of ImpactAssessmentMatrixProfile::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

3.3 (NC) CalculationBasedImpactAssessmentMatrix

Inheritance path = `ImpactAssessmentMatrix` : `IdentifiedObject`

Calculation based impact assessment matrix. It relates to the remedial action schedule.

Calculation-Based is the impact matrix determined by calculating the impact factors (eventually

scaled by the intensity of the remedial action) and matching them against a threshold in a determined way described by the methodologies.

Table 2 shows all attributes of CalculationBasedImpactAssessmentMatrix.

**Table 2 – Attributes of
ImpactAssessmentMatrixProfile::CalculationBasedImpactAssessmentMatrix**

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

3.4 (NC) ConnectingImpactAssessmentMatrix

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

Connecting system operator matrix is the impact matrix indicating which system operators are connecting for that specific remedial action. The concept of connecting system operator for a remedial action is defined by CSAm Article 2.1(14).

Table 3 shows all attributes of ConnectingImpactAssessmentMatrix.

**Table 3 – Attributes of
ImpactAssessmentMatrixProfile::ConnectingImpactAssessmentMatrix**

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

3.5 (abstract) IdentifiedObject root class

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

Table 4 shows all attributes of IdentifiedObject.

Table 4 – Attributes of ImpactAssessmentMatrixProfile::IdentifiedObject

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

3.6 (abstract,NC) ImpactAssessmentMatrix

Inheritance path = [IdentifiedObject](#)

The result of an impact assessment analysis for each remedial action or remedial action schedule onto the grid and operation of each system operator.
Table 5 shows all attributes of ImpactAssessmentMatrix.

Table 5 – Attributes of ImpactAssessmentMatrixProfile::ImpactAssessmentMatrix

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

3.7 (NC) ListBasedImpactAssessmentMatrix

Inheritance path = [ImpactAssessmentMatrix](#) : [IdentifiedObject](#)
List-Based is the impact matrix determined by agreement of the system operators involved. System operators jointly decide which Remedial Action (eventually scaled by the intensity of the remedial action) is impacting.
Table 6 shows all attributes of ListBasedImpactAssessmentMatrix.

**Table 6 – Attributes of
ImpactAssessmentMatrixProfile::ListBasedImpactAssessmentMatrix**

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

3.8 (abstract,NC) OutcomeValue root class

Outcome of an impact assessment matrix.
Table 7 shows all attributes of OutcomeValue.

Table 7 – Attributes of ImpactAssessmentMatrixProfile::OutcomeValue

name	mult	type	description
outcome	1..1	OutcomeImpactAssessmentKind	(NC) Outcome value.
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.

Table 8 shows all association ends of OutcomeValue with other classes.

Table 8 – Association ends of ImpactAssessmentMatrixProfile::OutcomeValue with other classes

mult from	name	mult to	type	description
1..*	ImpactAssessmentMatrix	1..1	ImpactAssessmentMatrix	(NC) the impact assessment matrix which has this value.

mult from	name	mult to	type	description
0..*	ImpactedSystemOperator	1..1	SystemOperator	(NC) The impacted system operator that has an outcome value.

3.9 (NC) OwnerRemedialActionAssessment root class

Owner remedial action assessment of the impact of their remedial action on neighboring system operators.

Table 9 shows all attributes of OwnerRemedialActionAssessment.

**Table 9 – Attributes of
ImpactAssessmentMatrixProfile::OwnerRemedialActionAssessment**

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.

Table 10 shows all association ends of OwnerRemedialActionAssessment with other classes.

**Table 10 – Association ends of
ImpactAssessmentMatrixProfile::OwnerRemedialActionAssessment with other classes**

mult from	name	mult to	type	description
0..*	RemedialActionImpact	1..1	RemedialActionImpact	(NC) Remedial action impact which is evaluated by the owner of the remedial action.
0..*	ImpactedSystemOperator	1..1	SystemOperator	(NC) System operator that is evaluated to be impacted by the remedial action done by the remedial action owner.

3.10 (NC) QualitativeRemedialActionImpact

Inheritance path = [RemedialActionImpact](#)

Defines the qualitative impact for a remedial action. Relevant remedial action is assumed to have impact when the impact quantity is applied.

Table 11 shows all attributes of QualitativeRemedialActionImpact.

**Table 11 – Attributes of
ImpactAssessmentMatrixProfile::QualitativeRemedialActionImpact**

name	mult	type	description
impactQuantity	0..1	Float	(NC) inherited from: RemedialActionImpact
kind	1..1	ImpactAgreementKind	(NC) inherited from: RemedialActionImpact
mRID	1..1	String	(NC) inherited from: RemedialActionImpact

Table 12 shows all association ends of QualitativeRemedialActionImpact with other classes.

**Table 12 – Association ends of
ImpactAssessmentMatrixProfile::QualitativeRemedialActionImpact with other classes**

mult from	name	mult to	type	description
0..*	RemedialAction	0..1	RemedialAction	(NC) inherited from: RemedialActionImpact
0..*	RemedialActionGroup	0..1	RemedialActionGroup	(NC) inherited from: RemedialActionImpact
0..*	AssessingSystemOperator	1..1	SystemOperator	(NC) inherited from: RemedialActionImpact

3.11 (NC) QuantitativeRemedialActionImpact

Inheritance path = [RemedialActionImpact](#)

Defines the quantitative impact for a remedial action. The value if the impact quantity is derived through offline calculation that has caused an impact of an element that is monitored by the assessed system operator higher than the relevant threshold for the conducting equipment. Table 13 shows all attributes of QuantitativeRemedialActionImpact.

**Table 13 – Attributes of
ImpactAssessmentMatrixProfile::QuantitativeRemedialActionImpact**

name	mult	type	description
impactQuantity	0..1	Float	(NC) inherited from: RemedialActionImpact
kind	1..1	ImpactAgreementKind	(NC) inherited from: RemedialActionImpact
mRID	1..1	String	(NC) inherited from: RemedialActionImpact

Table 14 shows all association ends of QuantitativeRemedialActionImpact with other classes.

**Table 14 – Association ends of
ImpactAssessmentMatrixProfile::QuantitativeRemedialActionImpact with other classes**

mult from	name	mult to	type	description
0..*	SensitivityArea	1..1	SensitivityArea	(NC) Sensitivity area which should be monitored to evaluate the threshold given by the remedial action impact on relevant equipment.
0..*	RemedialAction	0..1	RemedialAction	(NC) inherited from: RemedialActionImpact
0..*	RemedialActionGroup	0..1	RemedialActionGroup	(NC) inherited from: RemedialActionImpact
0..*	AssessingSystemOperator	1..1	SystemOperator	(NC) inherited from: RemedialActionImpact

3.12 (abstract,NC) RemedialAction root class

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.

3.13 (abstract,NC) RemedialActionImpact root class

Remedial action impact assessment based on a given agreement with a specific system operator.

Table 15 shows all attributes of RemedialActionImpact.

439 **Table 15 – Attributes of ImpactAssessmentMatrixProfile::RemedialActionImpact**

name	mult	type	description
impactQuantity	0..1	Float	(NC) Delta, positive or negative, quantity that when it is applied to the remedial action, it will cause impact on a conducting equipment monitored by the assessed system operator. Example of relevant remedial action changes are redispatching, countertrading, change of set point on HVDC systems or change of taps on phase-shifting transformers.
kind	1..1	ImpactAgreementKind	(NC) The impact agreement kind.
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.

440

441 Table 16 shows all association ends of RemedialActionImpact with other classes.

442 **Table 16 – Association ends of ImpactAssessmentMatrixProfile::RemedialActionImpact**
443 **with other classes**

mult from	name	mult to	type	description
0..*	RemedialAction	0..1	RemedialAction	(NC) The remedial action that has an impact.
0..*	RemedialActionGroup	0..1	RemedialActionGroup	(NC) The remedial action group that has an impact.
0..*	AssessingSystemOperator	1..1	SystemOperator	(NC) The impacted System Operator that assigns a remedial action impact.

444

445 **3.14 (NC) RemedialActionOutcomeValue**446 Inheritance path = [OutcomeValue](#)

447 Outcome of an impact assessment matrix for a remedial action.

448 Table 17 shows all attributes of RemedialActionOutcomeValue.

449 **Table 17 – Attributes of ImpactAssessmentMatrixProfile::RemedialActionOutcomeValue**

name	mult	type	description
impactQuantity	0..1	Float	(NC) Delta, positive or negative, quantity that when it is applied to the remedial action, it will cause impact on a conducting equipment monitored by the assessed system operator. Example of relevant remedial action changes are redispatching, countertrading, change of set point on HVDC systems or change of taps on phase-shifting transformers.
outcome	1..1	OutcomeImpactAssessmentKind	(NC) inherited from: OutcomeValue
mRID	1..1	String	(NC) inherited from: OutcomeValue

450

451 Table 18 shows all association ends of RemedialActionOutcomeValue with other classes.

**Table 18 – Association ends of
ImpactAssessmentMatrixProfile::RemedialActionOutcomeValue with other classes**

mult from	name	mult to	type	description
0..*	RemedialActionGroup	0..1	RemedialActionGroup	(NC) The remedial action group that has a remedial action group outcome value.
0..*	RemedialAction	0..1	RemedialAction	(NC) The remedial action that has a remedial action outcome value.
1..*	ImpactAssessmentMatrix	1..1	ImpactAssessmentMatrix	(NC) inherited from: OutcomeValue
0..*	ImpactedSystemOperator	1..1	SystemOperator	(NC) inherited from: OutcomeValue

3.15 (abstract,NC) RemedialActionSchedule root class

A schedule for a determined remedial action.

3.16 (NC) RemedialActionScheduleOutcomeValue

Inheritance path = [OutcomeValue](#)

Outcome of an impact assessment matrix for a remedial action schedule.

Table 19 shows all attributes of RemedialActionScheduleOutcomeValue.

**Table 19 – Attributes of
ImpactAssessmentMatrixProfile::RemedialActionScheduleOutcomeValue**

name	mult	type	description
outcome	1..1	OutcomeImpactAssessmentKind	(NC) inherited from: OutcomeValue
mRID	1..1	String	(NC) inherited from: OutcomeValue

Table 20 shows all association ends of RemedialActionScheduleOutcomeValue with other classes.

**Table 20 – Association ends of
ImpactAssessmentMatrixProfile::RemedialActionScheduleOutcomeValue with other classes**

mult from	name	mult to	type	description
0..*	RemedialActionSchedule	1..1	RemedialActionSchedule	(NC) The remedial action schedule that has a remedial action schedule outcome value.
1..*	ImpactAssessmentMatrix	1..1	ImpactAssessmentMatrix	(NC) inherited from: OutcomeValue
0..*	ImpactedSystemOperator	1..1	SystemOperator	(NC) inherited from: OutcomeValue

3.17 (abstract,NC) SensitivityArea root class

A monitoring area that defines the required observability area given by the sensitivity factors.

3.18 (abstract,NC) SystemOperator root class

System operator.

3.19 (NC) ImpactAgreementKind enumeration

The impact agreement for the remedial action.

476 Table 21 shows all literals of ImpactAgreementKind.

477 **Table 21 – Literals of ImpactAssessmentMatrixProfile::ImpactAgreementKind**

literal	value	description
noAgreement		No agreement is reached on the qualitative impact of a remedial action.
never		An agreement is reached that a remedial action is never impacting.
always		An agreement is reached that the remedial action is always impacting whichever the intensity.

478

479 **3.20 (NC) OutcomeImpactAssessmentKind enumeration**

480 Outcome impact assessments kinds.

481 Table 22 shows all literals of OutcomeImpactAssessmentKind.

482 **Table 22 – Literals of ImpactAssessmentMatrixProfile::OutcomeImpactAssessmentKind**

literal	value	description
true		True.
false		False.
undecided		Undecided. Used only for list-based impact assessment matrix.

483

484 **3.21 Float primitive**

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

486 **3.22 String primitive**

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

489

490

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

492 **A.1 General**

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

497 **A.2 Sample instance data**

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