



European Network of
Transmission System Operators
for Electricity

POWER SYSTEM PROJECT PROFILE SPECIFICATION

2024-10-16

ICTC APPROVED
VERSION 2.3.1

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
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.

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	10
45	2.4.2 Reference metadata	10
46	3 Detailed Profile Specification	10
47	3.1 General.....	10
48	3.2 (abstract,NC) AvailabilitySchedule root class	11
49	3.3 (NC) DifferenceModel	11
50	3.4 (NC) PowerSystemProject.....	12
51	3.5 (NC) PowerSystemProjectGroup root class	13
52	3.6 (abstract,dcat) Resource root class	14
53	3.7 Date primitive.....	15
54	3.8 DateTime primitive	15
55	3.9 Integer primitive	15
56	3.10 (profcim) IRI primitive.....	16
57	3.11 String primitive.....	16
58	3.12 (profcim) StringFixedLanguage primitive	16
59	3.13 (profcim) StringIRI primitive	16
60	3.14 (profcim) URL primitive	16
61	4 Appendix C: Use cases (informative)	16
62	4.1 Requirement	16
63	4.2 Use-Case Overview	17
64	4.3 IOP Test Cases	18
65	4.3.1 Purpose.....	18
66	4.3.2 TU PSP1: Cross border MAS projects.....	18
67	4.3.3 TU PSP02: Project Schedule Alternatives	21
68	4.3.4 TU PSP03: Project Lifecycle	24
69	4.3.5 TU PSP04: Power System Resource Variance Study	29
70	Annex A (informative): Sample data	30
71	A.1 General.....	30
72	A.2 Sample instance data.....	30
73		
74	List of figures	
75	Figure 1 – Class diagram PowerSystemProjectProfile::PowerSystemProjectProfile	11
76		

77 **List of tables**

78	Table 1 – Attributes of PowerSystemProjectProfile::DifferenceModel	12
79	Table 2 – Attributes of PowerSystemProjectProfile::PowerSystemProject	12
80	Table 3 – Association ends of PowerSystemProjectProfile::PowerSystemProject with	
81	other classes	13
82	Table 4 – Attributes of PowerSystemProjectProfile::PowerSystemProjectGroup	14
83	Table 5 – Attributes of PowerSystemProjectProfile::Resource	14
84		

85 1 Introduction

86 The document is describing power system project profile specification. The specification
87 includes profile definitions and test use cases.

88 Power system project profile provides metadata around the changes applied to a model and
89 exchanged using difference model exchange.

90 2 Application profile specification

91 2.1 Version information

92 The content is generated from UML model file CIM17-2_CGMES31v01_PROF-
93 20v02_NC23v65_MS10v01_DES10v01.eap.

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

- 95 - Title: Power System Project Vocabulary
- 96 - Keyword: PSP
- 97 - Description: This vocabulary is describing the power system profile.
- 98 - Version IRI: <https://ap-voc.cim4.eu/PowerSystemProject/2.3>
- 99 - Version info: 2.3.1
- 100 - Prior version:
- 101 - Conforms to: urn:iso:std:iec:61970-600-2:ed-1|urn:iso:std:iec:61970-301:ed-
102 7:amd1|file:///iec61970cim17v40_iec61968cim13v13a_iec62325cim03v17a.eap|urn:iso:
103 std:iec:61970-401:draft:ed-1|urn:iso:std:iec:61970-501:draft:ed-
104 2|file:///CIM100_CGMES31v01_501-20v02_NC23v62_MM10v01.eap
- 105 - Identifier: urn:uuid:29bfa45c-7d04-42f1-97c1-2e0f70f476a0

106 2.2 Constraints naming convention

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

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

110 where

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

112 rule.Standard: the number of the standard e.g. 301 for 61970-301, 456 for 61970-456, 13 for
113 61968-13. 61970-600 specific constraints refer to 600 although they are related to one or
114 combination of the 61970-450 series profiles. For NC profiles, NC is used.

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

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

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

121 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

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

- 171 • R:452:ALL:NA:associations

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

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

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

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

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

- 186 • R:452:ALL:NA:uniqueIdentifier

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

- 189 • R:452:ALL:NA:unitMultiplier

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

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

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

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

195 The string IdentifiedObject.description is maximum 256 characters.

- 196 • C:452:ALL:NA:float

197 An attribute that is defined as float (e.g. has a type Float or a type which is a Datatype
198 with .value attribute of type Float) shall support ISO/IEC 60559:2020 for floating-point
199 arithmetic using single precision floating point. A single precision float supports 7
200 significant digits where the significant digits are described as an integer, or a decimal
201 number with 6 decimal digits. Two float values are equal when the significant with 7
202 digits are identical, e.g. 1234567 is equal 1.234567E6 and so are 1.2345678 and
203 1.234567E0.

- 204 • R:NC:ALL:NA:serialization

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

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

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

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

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

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

- for association

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

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

```
</cim:PowerTransformer>
```

- for attribute

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

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

```
</cim:ACLineSegment>
```

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

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

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

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

```
</cim:Equipment>
```

2.4 Metadata

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

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

2.4.1 Constraints

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

- R:NC:ALL:wasAttributedTo:usage

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

2.4.2 Reference metadata

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

3 Detailed Profile Specification

3.1 General

This package contains power system project profile.

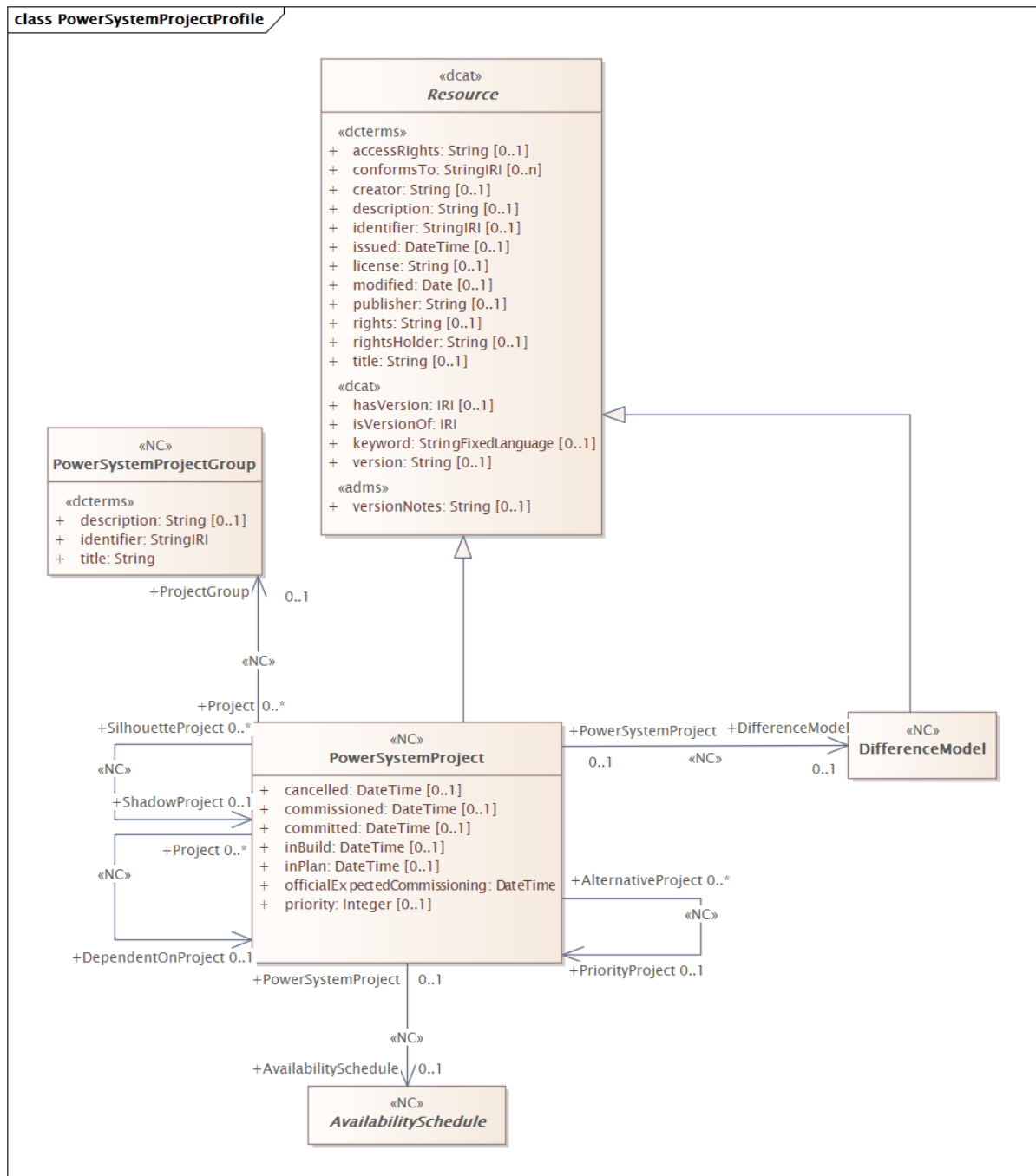


Figure 1 – Class diagram PowerSystemProjectProfile::PowerSystemProjectProfile

Figure 1: The diagram shows power system project profile.

3.2 (abstract,NC) AvailabilitySchedule root class

A given (un)availability schedule with a given status and cause that include multiple equipment that need to follow the same scheduling periods.

3.3 (NC) DifferenceModel

Inheritance path = [Resource](#)

A set of statements describing the changes in the network model. The statement is defined in the difference model.

289 Table 1 shows all attributes of DifferenceModel.

290 **Table 1 – Attributes of PowerSystemProjectProfile::DifferenceModel**

name	mult	type	description
accessRights	0..1	String	(dcterms) inherited from: Resource
conformsTo	0..n	StringIRI	(dcterms) inherited from: Resource
creator	0..1	String	(dcterms) inherited from: Resource
description	0..1	String	(dcterms) inherited from: Resource
hasVersion	0..1	IRI	(dcat) inherited from: Resource
identifier	0..1	StringIRI	(dcterms) inherited from: Resource
issued	0..1	DateTime	(dcterms) inherited from: Resource
isVersionOf	1..1	IRI	(dcat) inherited from: Resource
keyword	0..1	StringFixedLanguage	(dcat) inherited from: Resource
license	0..1	String	(dcterms) inherited from: Resource
modified	0..1	Date	(dcterms) inherited from: Resource
publisher	0..1	String	(dcterms) inherited from: Resource
rights	0..1	String	(dcterms) inherited from: Resource
rightsHolder	0..1	String	(dcterms) inherited from: Resource
title	0..1	String	(dcterms) inherited from: Resource
version	0..1	String	(dcat) inherited from: Resource
versionNotes	0..1	String	(adms) inherited from: Resource

291

292 3.4 (NC) PowerSystemProject

293 Inheritance path = [Resource](#)

294 Knowledge data for the power system project that describe the status and the planned
295 implementation of the changes into the as-built model.

296 Table 2 shows all attributes of PowerSystemProject.

297 **Table 2 – Attributes of PowerSystemProjectProfile::PowerSystemProject**

name	mult	type	description
cancelled	0..1	DateTime	From this date the project is in cancelled state. No further development will be done to the project or associated change set in this state.
commissioned	0..1	DateTime	From this date the project is in commissioned state. Any conducting equipment in the change set can be energized from this day. No further changes will be done to the change set.
committed	0..1	DateTime	From this date the project is in committed state. The change set will from this day be part of the as-build model.
inBuild	0..1	DateTime	From this day the project is in build state. Alternative project have been evaluated. Any procurement has started and the change set is being updated to an as-build model.
inPlan	0..1	DateTime	From this date the project is in planning state. Study or procurement strategy has triggered the start of a project involving changes to one or more models. Alternative projects and change sets are evaluated.

name	mult	type	description
priority	0..1	Integer	Priority between competing project. Use 0 for do not care. Use 1 for highest priority. Use 2 as priority is less than 1 and so on.
officialExpectedCommissioning	1..1	DateTime	Published official commissioning date.
accessRights	0..1	String	(dcterms) inherited from: Resource
conformsTo	0..n	StringIRI	(dcterms) inherited from: Resource
creator	0..1	String	(dcterms) inherited from: Resource
description	0..1	String	(dcterms) inherited from: Resource
hasVersion	0..1	IRI	(dcat) inherited from: Resource
identifier	0..1	StringIRI	(dcterms) inherited from: Resource
issued	0..1	DateTime	(dcterms) inherited from: Resource
isVersionOf	1..1	IRI	(dcat) inherited from: Resource
keyword	0..1	StringFixedLanguage	(dcat) inherited from: Resource
license	0..1	String	(dcterms) inherited from: Resource
modified	0..1	Date	(dcterms) inherited from: Resource
publisher	0..1	String	(dcterms) inherited from: Resource
rights	0..1	String	(dcterms) inherited from: Resource
rightsHolder	0..1	String	(dcterms) inherited from: Resource
title	0..1	String	(dcterms) inherited from: Resource
version	0..1	String	(dcat) inherited from: Resource
versionNotes	0..1	String	(adms) inherited from: Resource

Table 3 shows all association ends of PowerSystemProject with other classes.

Table 3 – Association ends of PowerSystemProjectProfile::PowerSystemProject with other classes

mult from	name	mult to	type	description
0..1	AvailabilitySchedule	0..1	AvailabilitySchedule	(NC) The availability schedule associated with this power system project.
0..1	DifferenceModel	0..1	DifferenceModel	(NC) The difference model describing this power system project.
0..*	ProjectGroup	0..1	PowerSystemProjectGroup	(NC) Power system project group to which this project belongs.
0..*	DependentOnProject	0..1	PowerSystemProject	(NC) Grouping of projects that are depending on each other. A project can only be linked to one dependent project.
0..*	ShadowProject	0..1	PowerSystemProject	(NC) A shadowing project that includes the same change set, but different timeline.
0..*	PriorityProject	0..1	PowerSystemProject	(NC) The project that has an alternative project.

3.5 (NC) PowerSystemProjectGroup root class

A container with project that are grouped together. Primarily used for navigation and to highlight the phases that an overall project can go through.

306 Table 4 shows all attributes of PowerSystemProjectGroup.

307 **Table 4 – Attributes of PowerSystemProjectProfile::PowerSystemProjectGroup**

name	mult	type	description
description	0..1	String	(dcterms) A free-text account of the resource. Description may include but is not limited to: an abstract, a table of contents, a graphical representation, or a free-text account of the resource.
identifier	1..1	StringIRI	(dcterms) A unique identifier of the resource being described or cataloged. The identifier might be used as part of the IRI of the resource, but still having it represented explicitly is useful. The identifier is a text string which is assigned to the resource to provide an unambiguous reference within a particular context.
title	1..1	String	(dcterms) A name given to the resource.

308

309 3.6 (abstract, dcat) Resource root class

310 Resource published or curated by a single agent.

311 Table 5 shows all attributes of Resource.

312 **Table 5 – Attributes of PowerSystemProjectProfile::Resource**

name	mult	type	description
accessRights	0..1	String	(dcterms) Information about who access the resource or an indication of its security status. Access Rights may include information regarding access or restrictions based on privacy, security, or other policies.
conformsTo	0..n	StringIRI	(dcterms) An established standard to which the described resource conforms.
creator	0..1	String	(dcterms) An entity responsible for making the resource. Recommended practice is to identify the creator with a URI. If this is not possible or feasible, a literal value that identifies the creator may be provided.
description	0..1	String	(dcterms) A free-text account of the resource. Description may include but is not limited to: an abstract, a table of contents, a graphical representation, or a free-text account of the resource.
hasVersion	0..1	IRI	(dcat) This resource has a more specific, versioned resource.
identifier	0..1	StringIRI	(dcterms) A unique identifier of the resource being described or cataloged. The identifier might be used as part of the IRI of the resource, but still having it represented explicitly is useful. The identifier is a text string which is assigned to the resource to provide an unambiguous reference within a particular context.
issued	0..1	DateTime	(dcterms) Date of formal issuance of the resource.

name	mult	type	description
			Recommended practice is to describe the date, date/time, or period of time as recommended for the property Date, of which this is a subproperty.
isVersionOf	1..1	IRI	(dcat) A related resource of which the described resource is a version, edition, or adaptation. Changes in version imply substantive changes in content rather than differences in format. This property is intended to be used with non-literal values. This property is an inverse property of hasVersion.
keyword	0..1	StringFixedLanguage	(dcat) A keyword or tag describing a resource.
license	0..1	String	(dcterms) A legal document under which the resource is made available. Recommended practice is to identify the license document with a URI. If this is not possible or feasible, a literal value that identifies the license may be provided.
modified	0..1	Date	(dcterms) Most recent date on which the item was changed, updated or modified. Recommended practice is to describe the date, date/time, or period of time as recommended for the property Date, of which this is a subproperty.
publisher	0..1	String	(dcterms) An entity responsible for making the resource available.
rights	0..1	String	(dcterms) A statement that concerns all rights not addressed with dct:license or dct:accessRights, such as copyright statements.
rightsHolder	0..1	String	(dcterms) Information about rights held in and over the resource. Typically, rights information includes a statement about various property rights associated with the resource, including intellectual property rights. Recommended practice is to refer to a rights statement with a URI. If this is not possible or feasible, a literal value (name, label, or short text) may be provided.
title	0..1	String	(dcterms) A name given to the resource.
version	0..1	String	(dcat) The version indicator (name or identifier) of a resource.
versionNotes	0..1	String	(adms) A description of changes between this version and the previous version of the resource.

313

314 **3.7 Date primitive**

315 Date as "yyyy-mm-dd", which conforms with ISO 8601. UTC time zone is specified as "yyyy-

316 mm-ddZ". A local timezone relative UTC is specified as "yyyy-mm-dd(+/-)hh:mm".

317 **3.8 DateTime primitive**

318 Date and time as "yyyy-mm-ddThh:mm:ss.sss", which conforms with ISO 8601. UTC time zone

319 is specified as "yyyy-mm-ddThh:mm:ss.sssZ". A local timezone relative UTC is specified as

320 "yyyy-mm-ddThh:mm:ss.sss-hh:mm". The second component (shown here as "ss.sss") could

321 have any number of digits in its fractional part to allow any kind of precision beyond seconds.

322 **3.9 Integer primitive**

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

3.10 (profcim) IRI primitive

An IRI (Internationalized Resource Identifier) within an RDF graph is a Unicode string that conforms to the syntax defined in RFC 3987.

The primitive is serialized as `rdf:resource` in RDFXML.

IRIs in the RDF abstract syntax must be absolute, and may contain a fragment identifier.

IRI equality: Two IRIs are equal if and only if they are equivalent under Simple String Comparison according to section 5.1 of [RFC3987]. Further normalization must not be performed when comparing IRIs for equality.

IRIs are a generalization of URIs [RFC3986] that permits a wider range of Unicode characters. Every absolute URI and URL is an IRI, but not every IRI is an URI. When IRIs are used in operations that are only defined for URIs, they must first be converted according to the mapping defined in section 3.1 of [RFC3987]. A notable example is retrieval over the HTTP protocol. The mapping involves UTF-8 encoding of non-ASCII characters, %-encoding of octets not allowed in URIs, and Punycode-encoding of domain names.

3.11 String primitive

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

3.12 (profcim) StringFixedLanguage primitive

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

The primitive is serialized as literal without language support.

3.13 (profcim) StringIRI primitive

An IRI (Internationalized Resource Identifier) within an RDF graph is a Unicode string that conforms to the syntax defined in RFC 3987.

The primitive is serialized as literal without language support.

IRIs in the RDF abstract syntax must be absolute, and may contain a fragment identifier.

IRI equality: Two IRIs are equal if and only if they are equivalent under Simple String Comparison according to section 5.1 of [RFC3987]. Further normalization must not be performed when comparing IRIs for equality.

IRIs are a generalization of URIs [RFC3986] that permits a wider range of Unicode characters. Every absolute URI and URL is an IRI, but not every IRI is an URI. When IRIs are used in operations that are only defined for URIs, they must first be converted according to the mapping defined in section 3.1 of [RFC3987]. A notable example is retrieval over the HTTP protocol. The mapping involves UTF-8 encoding of non-ASCII characters, %-encoding of octets not allowed in URIs, and Punycode-encoding of domain names.

3.14 (profcim) URL primitive

A Uniform Resource Locator (URL), colloquially termed a web address, is a reference to a web resource that specifies its location on a computer network and a mechanism for retrieving it. A URL is a specific type of Uniform Resource Identifier (URI), although many people use the two terms interchangeably. URLs occur most commonly to reference web pages (http), but are also used for file transfer (ftp), email (mailto), database access (JDBC), and many other applications.

4 Appendix C: Use cases (informative)

4.1 Requirement

Difference model exchange is used to describe the changes to the network model. Additional meta-data is necessary to describe the content of the changes.

The additional meta-data MUST give information on:

- Date and time the changes will be added to the network model.

- State the realization of the changes is. A minimum would be to separate in planning, in build, cancelled and commissioned changes.
- Date and time the new equipment would be in of service. Updated and deleted equipment will follow standard outage handling.
- Grouping of set of changes that belongs together but do not have the same commissioned data, e.g. a project that consist of depended subprojects that will be commissioned before the main project (that has been approved by the government).
- Version on the changes. It is not practical to support tacking on changes on changes.
- Necessary references to a particular situation (EQ, SSH, TP, SV). In some cases a project is only commissioned (including in a planning model) in a certain situation.
- Dependency between projects. In case on alternative solutions of a problem.

The additional meta-data SHOULD give information on:

- Date and time the changes was actual added to the network model.
- Date and time the changes was cancelled.
- Mutual excluded changes. Alternative dependent changes that will NOT be implemented together.
- Priority of alternative changes.

The additional meta-data COULD give information on:

- Type of changes. A classification that would help systems on an information bus to identify changes that are relevant for their system without investigating all the detail changes.
- Model responsible. The organization that are primary responsible for the model.

4.2 Use-Case Overview

The use-case for the meta-data for changes are relevant for all the use-cases that changes to the model is included.

A change set or group of change sets will in a given organization be created on one of more of the following cases:

1. Created by the network analysing team (System Development Planning, Protection Planning).
2. Construction project (new or maintenance).
3. External organization.

An organisation could have one or more system that manages change set. A normal minimum would be a System Development Planning tool and an Operation (EMS/DMS). In many organisations the same changes are modelled in parallel for both systems. The CIM standard need to support the possibility to model the change set once and reuses it both for System Development Planning and Operation.

In the planning phase there could be multiple systems that are contributing with analysis and modelling on a given change set. It must be possible to exchange the set and the additional information between the systems. The following exchanges are relevant:

- System Development Planning
- Protection Planning
- Market design and planning
- Asset Construction
- TSO/DSO (DNO), TSO/TSO, DSO (DNO)/DSO (DNO)
- Government, regional/European (ENTSO-E)
- Research projects challenging different solutions (minor impact in terms of exchanges)

As construction is finalising, the “as build” model need to be exchange and in some cases contributes by the following system and business function:

- Outage Management/Scheduling System (OMS/OSS)
- Market Management System (MMS)
- Operation Planning and Operation (EMS/DMS)
- Grid Settlement
- Data Warehouse, measurement historian

4.3 IOP Test Cases

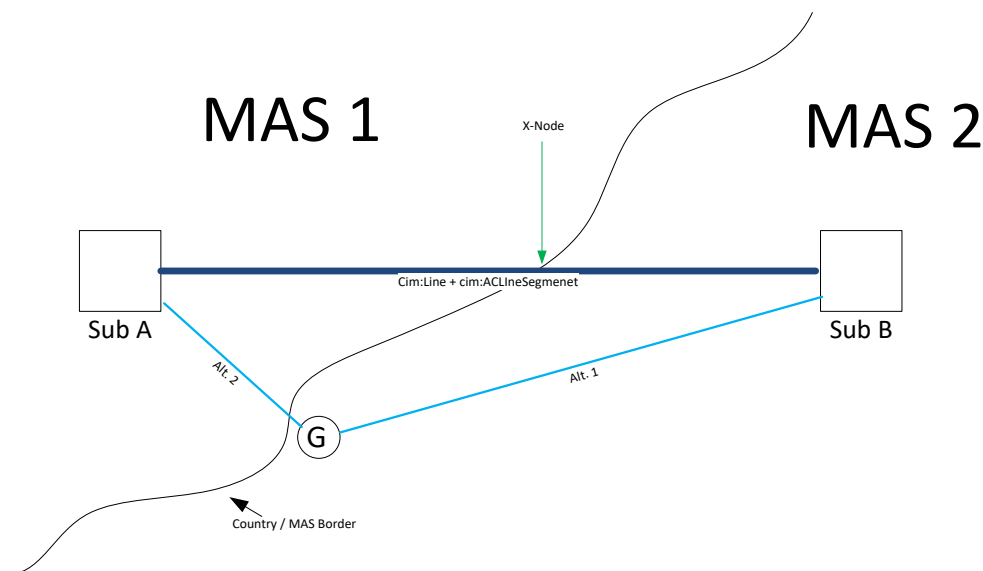
4.3.1 Purpose

The aim of this test is to demonstrate the ability of the tool to administrate Power System Project (PSP). The test will verifying the following functionality:

- Add changes/increment to a Power System Project
- Change attributes including dates on the project
- Change status of a project
- Split a project into two depending projects
- Create competing project
- The following main use cases are covered:
- Cross border MAS projects
- Project Schedule Alternatives
- Project Lifecycle

4.3.2 TU PSP1: Cross border MAS projects

The test handles the exchange of changes to models that represent the construction of a new line between to existing substation that is located in two separate MAS. The Boundary point needs to be defined and to different users should make the construction separate and on a later stage merge. Two alternative connection of a generating unit should be evaluated.

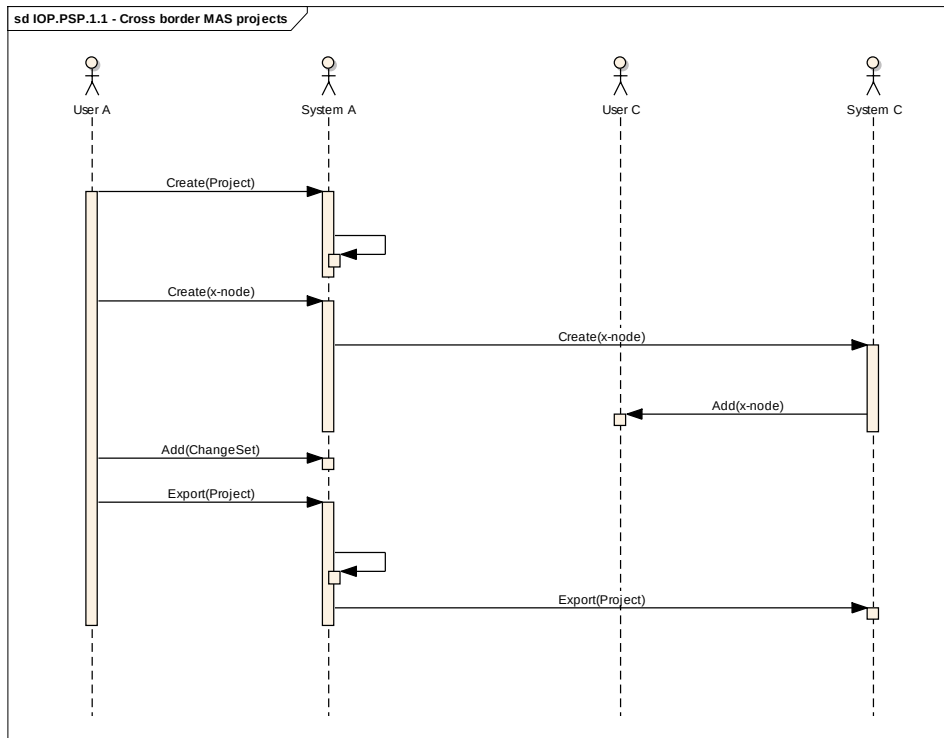


Three actors are involved:

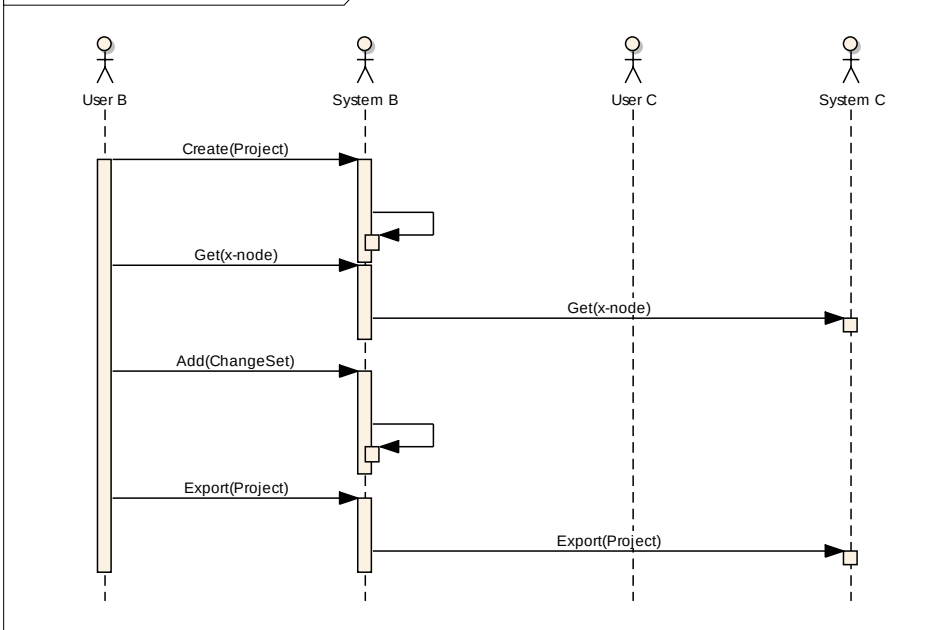
- User U_A with system S_A are updating MAS_1
- User U_B with system S_B are updating MAS_2
- User U_C with system S_C are merging change in MAS_1 with changes in MAS_2 and make dependent change to both MAS.

The sequence of steps is the following:

- User U_A through system S_A create (or request) a Boundary point from system S_C
- User U_A through system S_A build the ACLineSegment up to the Boundary Point
- User U_A through system S_A transfer the change set to System S_C
- User U_B using system S_B download the Boundary point information from system S_C
- User U_B build the ACLineSegment from the Boundary point
- User U_B through system S_B transferees the change set to System S_C
- User U_C through system S_C are merging the change set
- User U_C though system S_C are providing power flow study that covers both MAS_1 and MAS_2 including both change set
- User U_C through system S_C are making change two changes, add generator G to substation A in MAS_1 , and add generator G to substation B in MAS_2 . These changes are mutual excluded. One of the changes should be included in a given study.
- User U_C through system C is transferring all changes to both system S_A and system S_B
- User U_A through system S_A and User U_B through system S_B are calculating two cases for both MAS_1 and MAS_2 including all changes. Each of the case will be based on the mutual excluded changes
- User U_A through system S_A and User B through system S_B are transferring there result to system S_C .

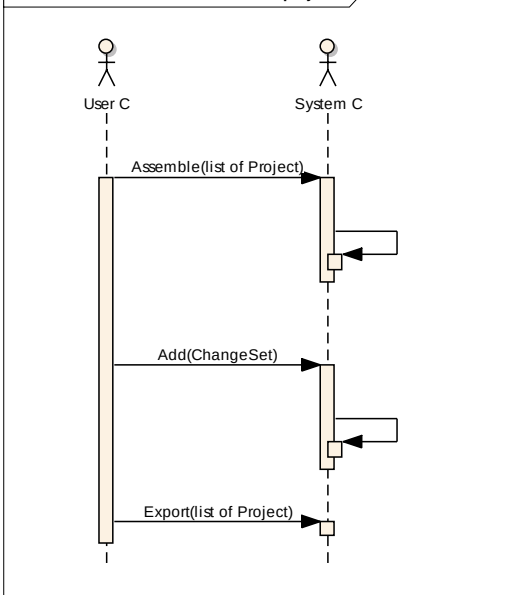


sd IOP.PSP.1.2 - Cross border MAS projects

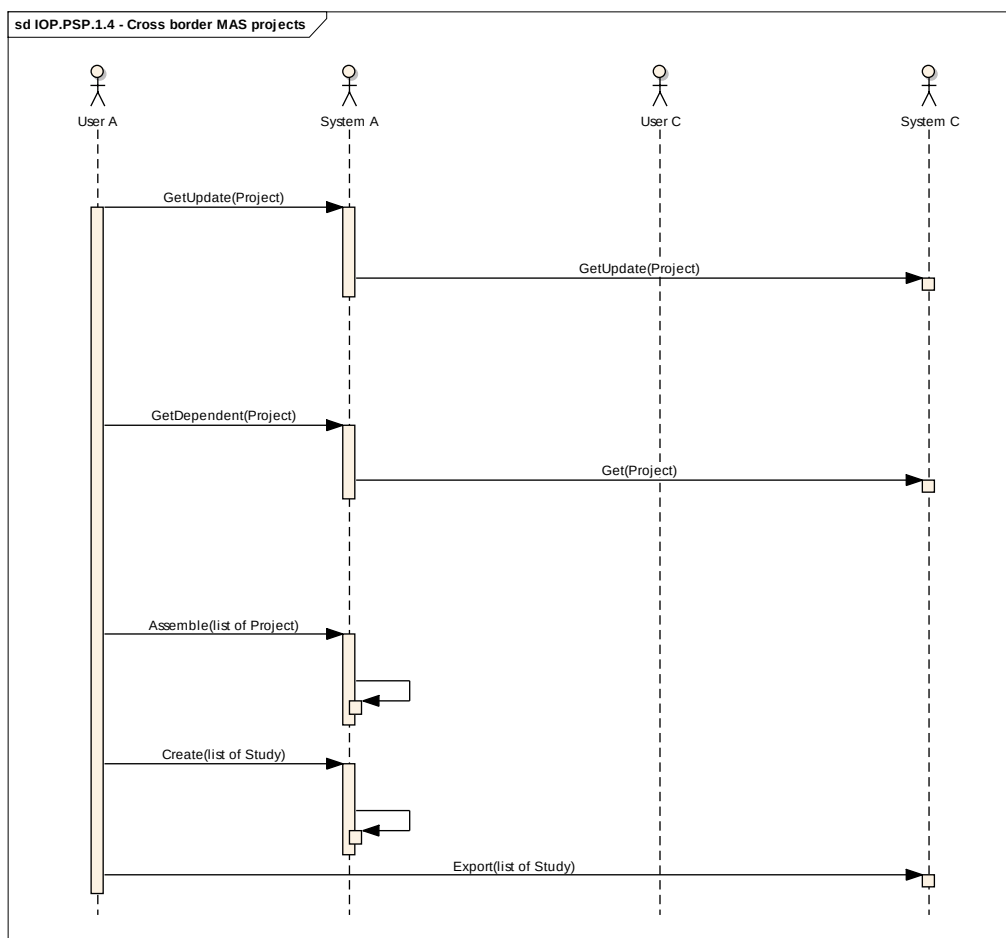


474

sd IOP.PSP.1.3 - Cross border MAS projects



475



User B with System B does the same.

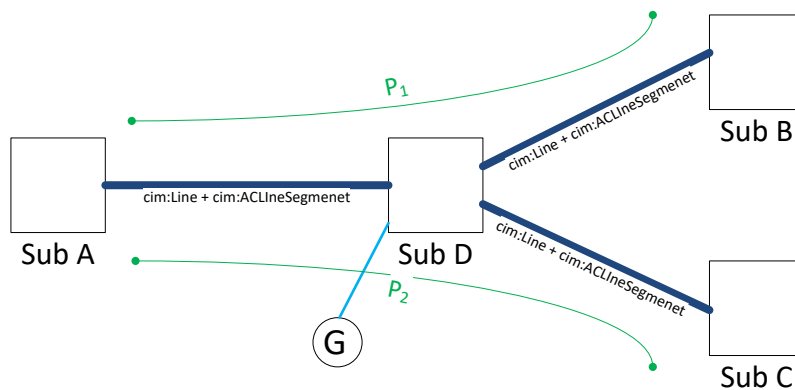
4.3.3 TU PSP02: Project Schedule Alternatives

There is a need to connect a major generating unit, G, to the network. Through analysing studies it is found that it should be connected together with the construction of substation D. However, it not yet decided if this should be done through project P₁ or project P₂.

Project P₁ include the construction of ACLineSegment_{AD}, substation D, generating unit G connected to substation D and ACLineSegment_{DB}.

Project P₂ include the construction of ACLineSegment_{AD}, substation D, generating unit G connected to substation D and ACLineSegment_{DC}.

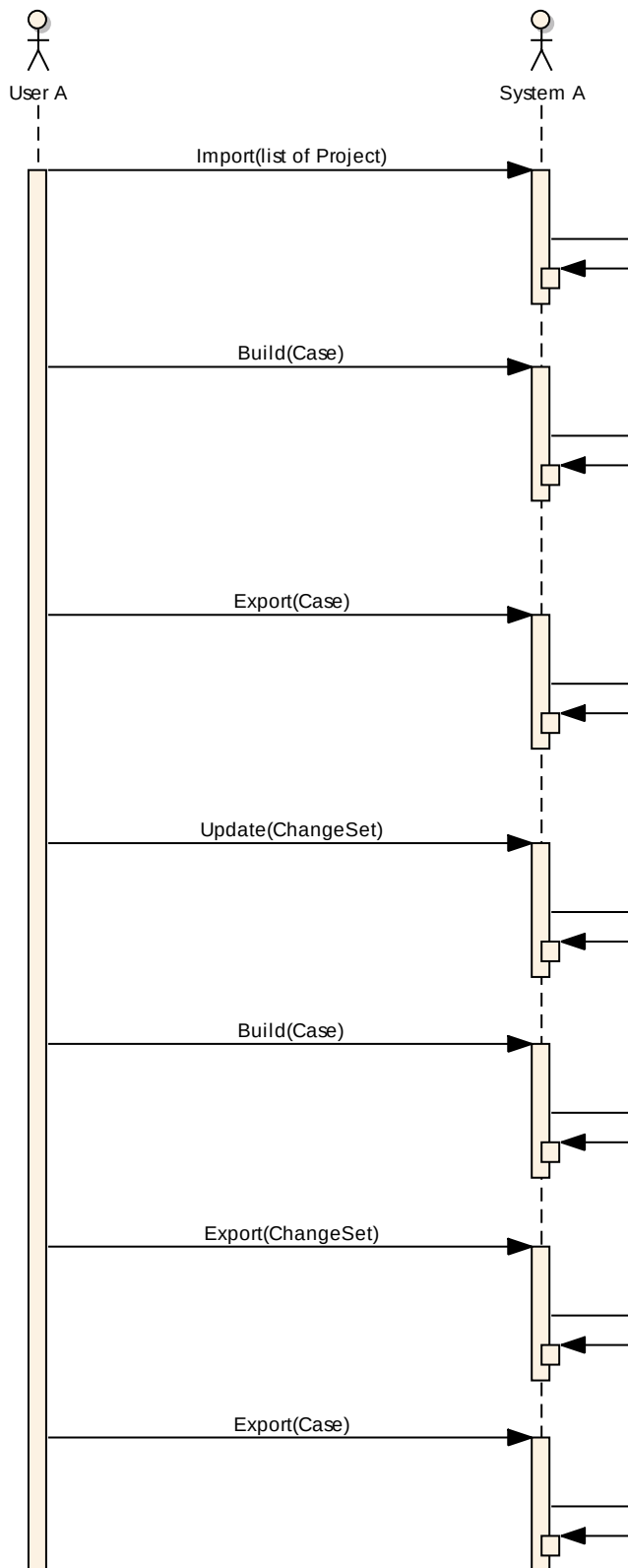
Project P₁ and project P₂ are competing project (mutually excluded). Both projects include the construction of ACLineSegment_{AD}, substation D and generating unit G connected to substation D. However, in project P₁ are the construction of ACLineSegment_{AD} required before energising the generator. In project P₂ are this construction scheduled after the energising.



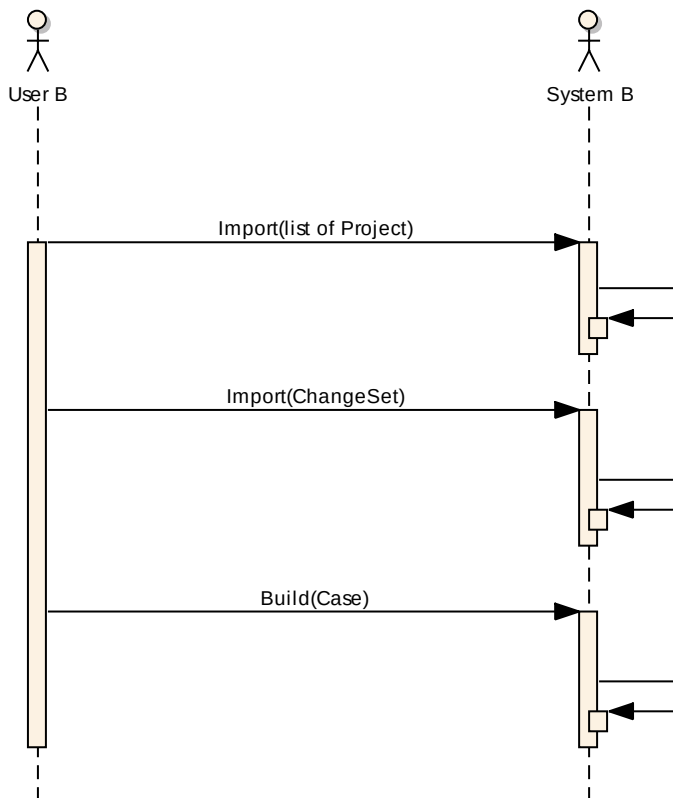
The sequence of steps is the following:

- User U_A through system S_A imports project P_1 and P_2
- User U_A through system S_A build and calculate a case including P_1 and export the results
- User U_A through system S_A build and calculate two case based on P_2 and export the results. One case does not include the $ACLineSegment_{AD}$
- User U_A through system S_A change one parameters $ACLineSegment_{AD}$. This one change should be reflected in both project P_1 and P_2 .
- User U_A through system S_A build and calculate cases C_1 including P_1 and case C_2 including P_2 .
- User U_A through system S_A exports the change and the result from the case C_1 and case C_2 .
- User U_B through system S_B imports project P_1 and P_2
- User U_B through system S_B imports the change done by user U_A through system S_A and calculate the case C_1 and case C_2 .

sd IOP.PSP.2.1 - Project Schedule Alternatives



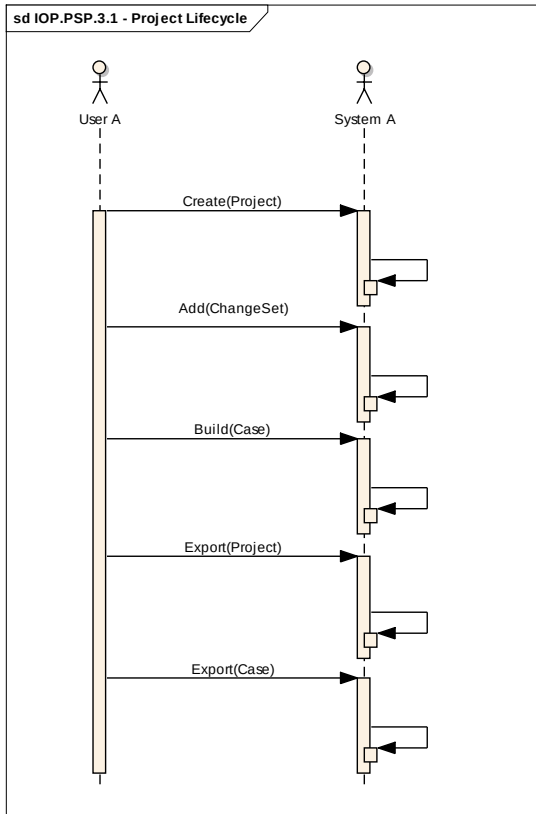
sd IOP.PSP.2.2 - Project Schedule Alternatives



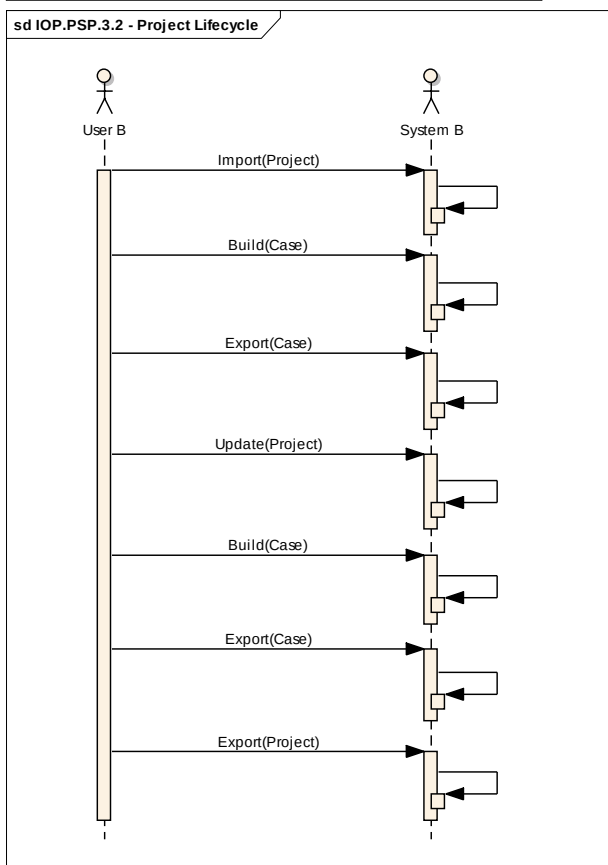
507
508
509

510 4.3.4 TU PSP03: Project Lifecycle

511 The developments of a project lifecycle from an analytic case to a commission the changes. The project
512 evolution is starting with one project that is split into subproject with more detail in the change set. The
513 update to the project is alternating between two users and with two different systems.



514



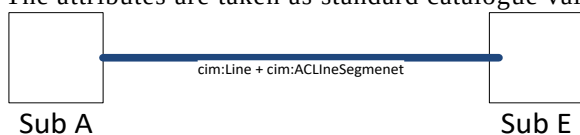
515

516

517

Stage 1. Early Analysis Stage

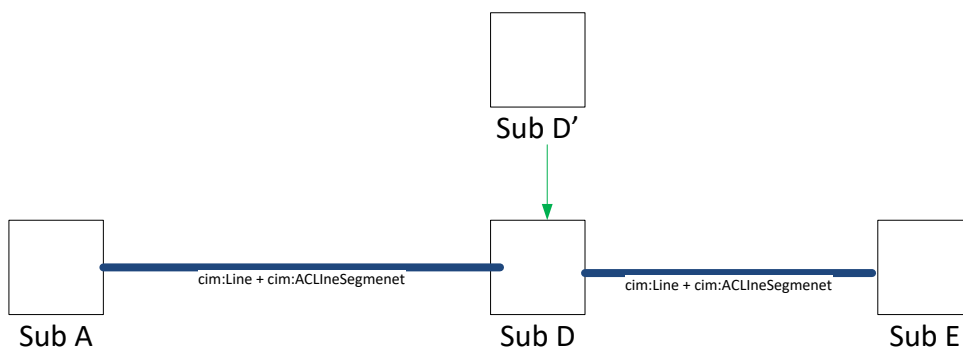
518 This stage is focused on low detailing and many alternative projects or solutions to a given problem.
 519 Exchanges of alternative project are covered in another test use case. The project is created with a
 520 change set including the addition of an ACLineSegment, ca. 150 km duplex parrot, between two existing
 521 substation, Sub A and Sub E. Add Transformer of 300 MVA to substation Sub E.
 522 The attributes are taken as standard catalogue values.



523
 524 The following analysis/operation should be able to be done:
 525 • Power Flow solution including the new line segment
 526 • Short-circuit level (balanced 3 phase)
 527 • Display diagram including the new line segment
 528 • Rough geographical location information
 529 • Positive system model adequate (including dynamic model)

530 531 Stage 2. First Public Analysis Stage

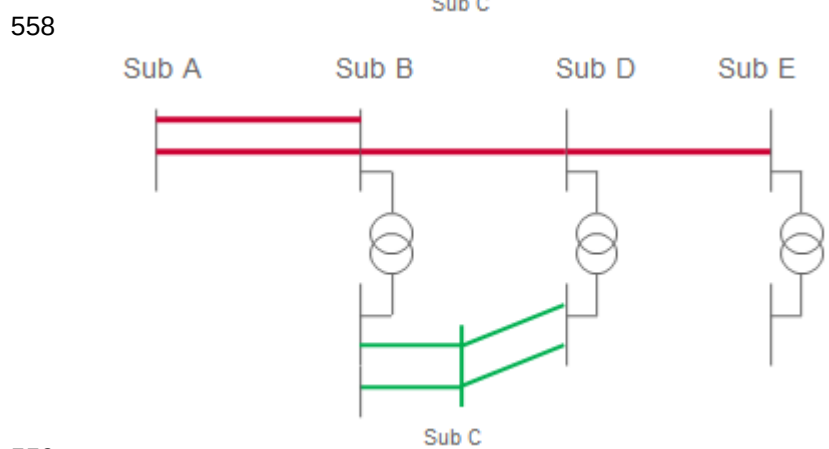
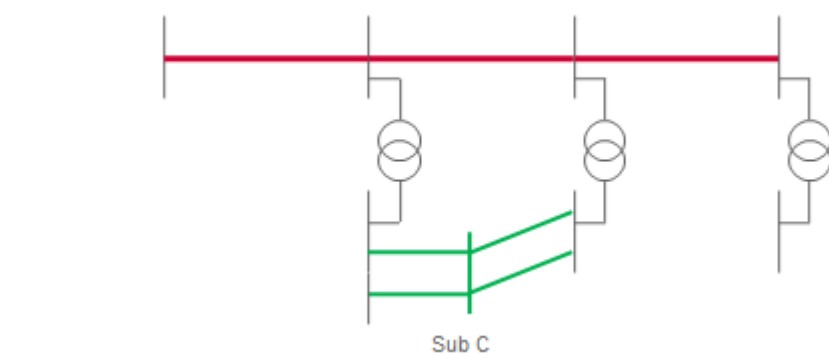
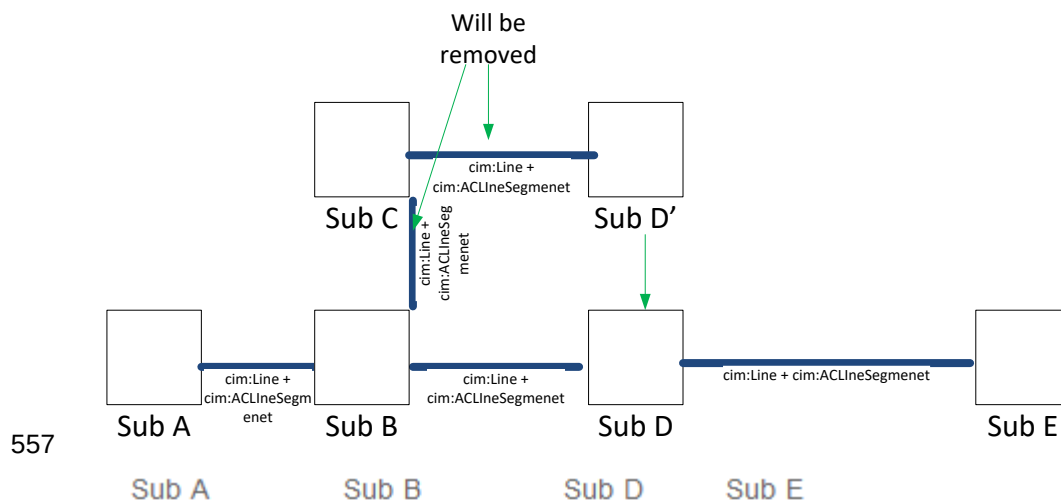
532 The alternatives are limited, but existing. The key point is to find out as much as possible to evaluate
 533 the "best" alternative from the previous stage. The output of this stage can be used as requirement for
 534 Request for Proposal (RFP) for the project and the equipment acquisition.
 535 Exchanges of alternative project are covered in another test use case.
 536 It is now clear that an existing substation, Sub D', needs to be "moved" and enhanced to become Sub D.
 537 The exact schedule does not need to be defined, but sufficient staging with use of temporarily
 538 configuration needs to be defined so that a reasonable confidence of the approached will work.
 539

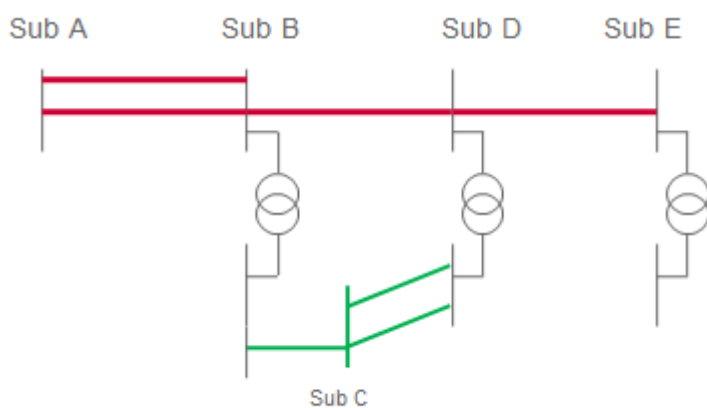


540
 541 The following analysis/operation should be able to be done:
 542 • Power Flow solution including the new line segment
 543 • Short-circuit level (balanced 3 phase)
 544 • Earth fault /Single phase short circuit calculations (1-phase faults)
 545 • Display diagram including the new line segment
 546 • Rough geographical location information
 547 • Positive and zero sequence system model necessary (including dynamic model)

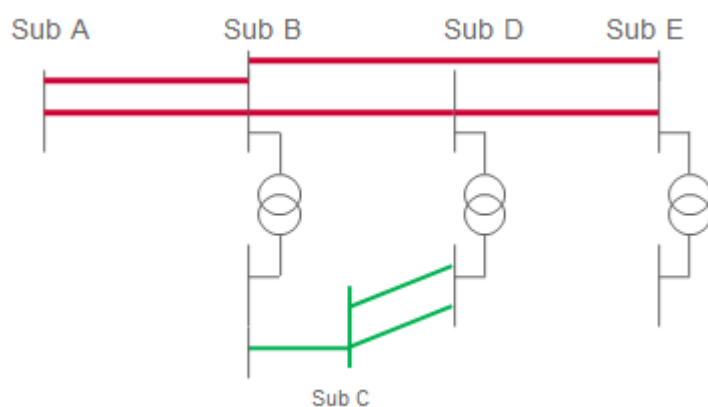
548 549 Stage.3. Application of License Stage

550 At this stage there is normally only one alternative. This model is updated to reflect the chosen project
 551 vendor and equipment. The detail needs to be so that each operating stage can be analyzed in detail.
 552 Standard catalogue values are replaced with vendor catalogue values.
 553 It is now clear that the line will go by substation Sub B. This will trigger that the line between Sub B
 554 and Sub C will be removed after Sub A and Sub B are connected. The full removed can only be done
 555 when Sub D is "moved" and connected the line already constructed from Sub B and Sub E.
 556 The sequences of the subproject are described in the diagram below.

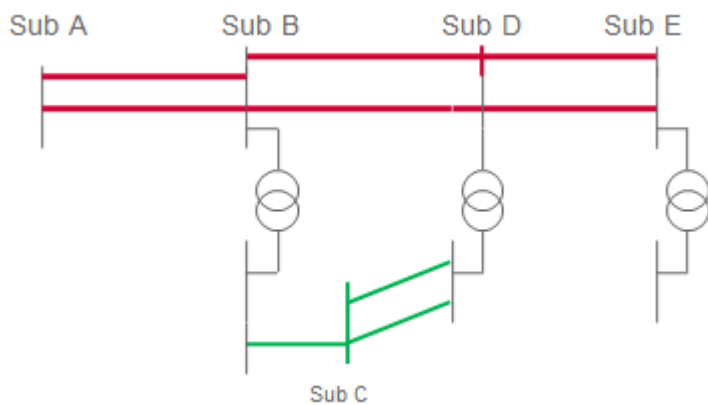




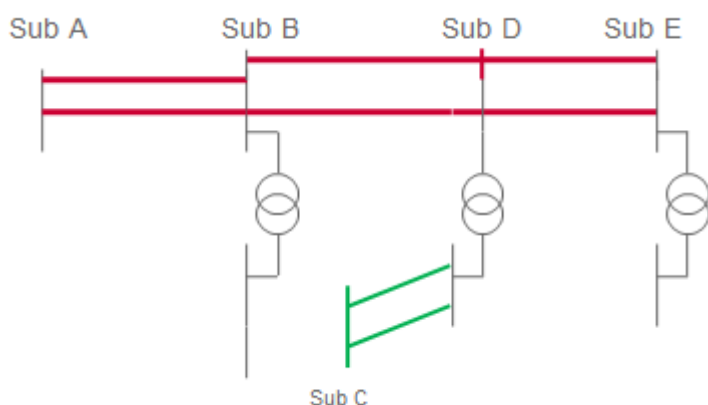
560



561



562



The following analysis/operation should be able to be done:

- Power Flow solution including the new line segment
- Short-circuit level (balanced 3 phase)
- Protection planning and fault analysis (all fault types)
- Display diagram including the new line segment
- Accurate geographical location information
- Positive and zero sequence system model necessary (including dynamic model)

Stage.4. Build/Construction Stage

The project is updated by subproject are update with detail separately. Vendor catalogue values are replaced with actual "measured" values. Part of the project is committed to the base model as-build model. Breaker information, measurement and control information are added to support an EMS model. Only additional breaker information will be part of this test case.

The subproject (or full project) that includes the "as-build" information will be imported to a State Estimate based system (EMS).

Planned outage, switch plan and EMS based analysis are not included in this test case.

Stage.5. Operational Stage

The changes are added to the base model. Operations of the added equipment are controlled by planned outage. Changes to the model are handled as new change set rather than update to existing change set. Planned outage, switch plan and EMS based analysis in addition to fault analysis including protection are not included in this test case.

4.3.5 TU PSP04: Power System Resource Variance Study

A System Development Planning study normally start with a given future base model. This model will include the collection of most probabilistic changes up to the data of Power System Resource variance Study.

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

596 **A.1 General**

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

601 **A.2 Sample instance data**

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

603

604