SEAS

Smart Energy Aware Systems
ITEA2 –– 12004

D2.2 SEAS Knowledge Model

Author of Deliverable: Maxime Lefrançois (ARMINES-Fayol)

With contributions from: Jarmo Kalaoja (VTT), Takoua Ghariani (IMT), Antoine Zimmermann (ARMINES Fayol)

Quality reviewers: Rémi Pecqueur (ENGIE), Lynda Témal (ENGIE)

Deliverable nature:	**Report(R)**
Dissemination level: (Confidentiality)	**Public (PU)**
Contractual delivery date:	**20 December 2016**
Actual delivery date:	**20 December 2016**
Version:	**1.0**
Total number of pages:	**76**
Keywords:	**Ontology, Knowledge model, Semantic Web, OWL, Semantic interoperability**

Abstract

This deliverable concentrates on the results of task 2.2 of work package 2.

It describes the SEAS Knowledge Model as a basis for semantic interoperability in the SEAS ecosystem. The SEAS Knowledge Model consists of an innovative Web ontology that is designed to: (i) meet the current best practices in terms of quality, metadata, and publication, (ii) reuse or align to existing standards, and (iii) cover the required expressivity for the SEAS use cases, while being extensible to other use cases and domains (gas, water, air, waste management).

This document is a snapshot of the situation at the end of the SEAS project. Up-to-date information can be found at the following websites:

https://w3id.org/seas/ for the SEAS Knowledge Model, and contributing to it;

https://w3id.org/pep/ for the Process Executor Platform ontology.

Executive Summary

Environmental, economic and sustainability challenges of continuously increasing energy consumption are present all over the world. SEAS (Smart Energy Aware Systems) project addressed the problem of inefficient and unsustainable energy consumption, which is due to a lack of sufficient means to control, monitor, estimate and adapt energy usage of systems in response to the dynamic situations and circumstances that influence them.

Another problem the project is tackling is that consumption related information is scattered across many standards and metrics, and consequently, there is no convenient way to engineer solutions that require interworking and exchanging energy information across related industries, standards and metrics.

This document first describes the SEAS Knowledge Model as a basis for semantic interoperability in the SEAS ecosystem. The SEAS Knowledge Model consists of an innovative Web ontology that is designed to: (i) meet the current best practices in terms of quality, metadata, and publication, (ii) reuse or align to existing standards, and (iii) cover the required expressivity for the SEAS use cases, while being extensible to other use cases and domains (gas, water, air, waste management).

This document is a snapshot of the situation at the end of the SEAS project. Up-to-date information can be found at the following websites:

https://w3id.org/seas/ for the SEAS Knowledge Model, and contributing to it;

https://w3id.org/pep/ for the Process Executor Platform ontology.

Document Information

Project Number	ITEA2 – 12004	**Acronym**	SEAS
Full Title	Smart Energy Aware Systems		
Project URL	http://the-smart-energy.com/		
Document URL	ITEA 3 · Project · 12004 SEAS		
ITEA Project Officer	Philippe Letellier		

Deliverable	**Number**	D2.2	**Title**	SEAS Knowledge Model
Work Package	**Number**	WP2	**Title**	Knowledge and Service Engineering

Date of Delivery	**Contractual**	05/01/2017	**Actual**	05/01/2017
Status	Version 1.0		final ☑	
Nature	prototype ☐ report ☑ dissemination ☐			
Dissemination level	public ☑ consortium ☐			

Authors (Partner)	Maxime Lefrançois (ARMINES Fayol), Jarmo Kalaoja (VTT), Takoua Ghariani (IMT), Antoine Zimmermann (ARMINES Fayol)			
Responsible Author	**Name**	Maxime Lefrançois	**E-mail**	maxime.lefran-cois@emse.fr
	Partner	ARMINES Fayol	**Phone**	+33477426640

Abstract (for dissemination)	This deliverable concentrates on the results of task 2.2 of work package 2. It describes the SEAS Knowledge Model as a basis for semantic interoperability in the SEAS ecosystem. The SEAS Knowledge Model consists of an innovative Web ontology that is designed to: (i) meet the current best practices in terms of quality, metadata, and publication, (ii) reuse or align to existing standards, and (iii) cover the required expressivity for the SEAS use cases, while being extensible to other use cases and domains (gas, water, air, waste management). This document is a snapshot of the situation at the end of the SEAS project. Up-to-date information can be found at the following websites: https://w3id.org/seas/ for the SEAS Knowledge Model, and contributing to it; https://w3id.org/pep/ for the Process Executor Platform ontology.
Keywords	Ontology, Knowledge model, Semantic Web, OWL, Semantic interoperability

Version Log

Issue Date	Rev. No.	Author	Change
20/05/2016	V0.0	Maxime Lefrançois (Armines)	Reuse D 2.2.1 as draft for D 2.2.
24/05/2016	V0.1	Maxime Lefrançois (Armines)	Adding specification for the ontology server
19/09/2016	V0.2	Jarmo Kalaoja (VTT)	Revised document, added chapter about online documentation and examples of knowledge model use with SEAS architecture.
25/11/2016	V0.3	Maxime Lefrançois (Armines)	Updated methodology section Updated all ontology report sections Updated use section
05/12/2016	V0.4	Maxime Lefrançois (Armines)	Changed the template, and switched to LaTeX
18/12/2016	V0.5	Maxime Lefrançois (Armines)	Added attributions to each section, Added work about SPARQL-Generate Added work about on-the-fly support of datatypes Added work about RDFP
20/12/2016	V0.6	Maxime Lefrançois (Armines)	Finalized abstract, executive summary, introduction, conclusion
24/12/2016	V0.7	Maxime Lefrançois (Armines)	Corrections proposed by Lynda Temal (ENGIE)
05/01/2017	V1.0	Maxime Lefrançois, Antoine Zimmermann (Armines)	Corrections proposed by Rémi Pecqueur (ENGIE), and deletion of sections about SPARQL-Generate, the on-the-fly support of datatypes and RDFP

Table of Contents

List of Figures

List of Tables

Abbreviations

SEAS	Smart Energy Aware Systems
OWL	Web Ontology Language
URL	Uniform Resource Locator
URI	Uniform Resource Identifier
IRI	Internationalized Resource Identifier
RDF	Resource Description Framework
DL	Description Logics
SPARQL	SPARQL Protocol and RDF Query Language
HTML	Hypertext Markup Language
XML	Extensible Markup Language
JSON	JavaScript Object Notation
SSN	Semantic Sensor Network
RDFS	RDF Schema
SAREF	Smart Appliances REFerence ontology
CSV	Comma-separated values
VOWL	Visual Notation for OWL Ontologies
EV	Electric Vehicle
EVSE	Electric Vehicle Service Equipment
HEMS	Home Energy Management System
UML	Unified Modeling Language
JSON-LD	JSON for Linked Data

1 Introduction

The SEAS knowledge model is a key enabler for the semantic interoperability at the basis of SEAS use cases and business models. The SEAS Knowledge Model, along with the SEAS Architecture, enables SEAS communicating nodes to expose, exchange, reason, and query knowledge in a semantically interoperable manner.

Figure 1 represents the traditional data-information-knowledge-wisdom pyramid that shows the hierarchical levels of understanding. The SEAS knowledge model belongs to the third level of understanding.

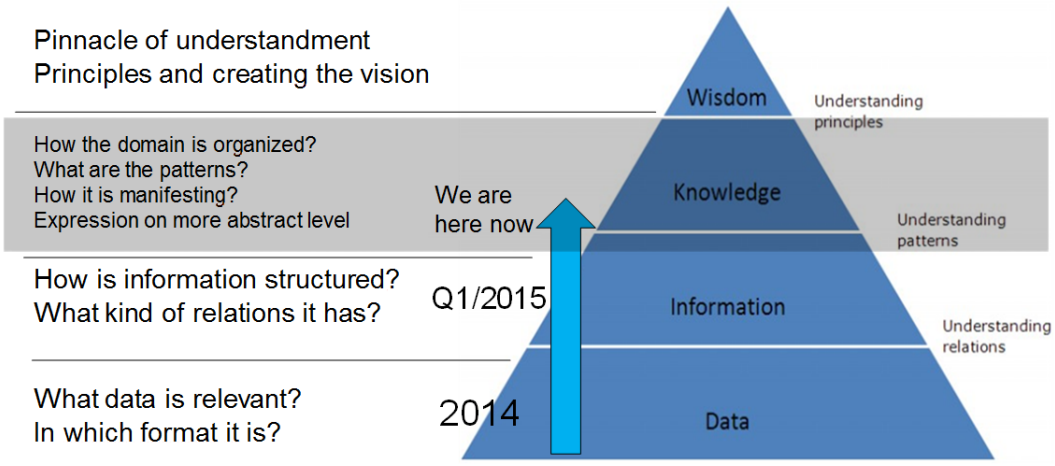

Figure 1: Traditional Data-Information-Knowledge-Wisdom (DIKW) pyramid with added data of SEAS

The SEAS knowledge model consists of a set of OWL2 DL ontologies, or *ontology modules*. Section 2 justifies the general requirements and design choices that guided the SEAS ontologies modeling task, the development of quality checker tools, and the development of the website that exposes the ontologies on the Web following the current best practices in terms of quality and metadata.

Then, Section 3 presents the different methodologies that have been applied to design the Knowledge Model.

Section 4 gives an informal description of each concept in the SEAS ontologies. The core modules define some basic design patterns that are instantiated in other modules. These modules cover the required expressivity for the SEAS use cases, and other could be added to adapt it to other use cases and domains (gas, water, air, waste management). The SEAS ontologies reuse or are align to some existing standards.

This work led to publications in Semantic Web conferences, and to open-source software development reusable by third parties.

2 Requirements and design choices for the ontologies and their publication (ARMINES-Fayol)

The SEAS knowledge model is a modularized and versioned OWL 2 DL ontology. Great effort has been made in the Semantic Web community to define best practices for the definition and the publication of ontologies. This section summarizes these practices, and justifies general requirements and design choices that guided the SEAS ontologies modeling task, the development of quality checker tools, and the development of the website that exposes the ontologies on the Web.

First, Section 2.1 lists general requirements the ontologies need to satisfy. Then, Section 2.5 then describes how the SEAS ontologies are related to other ontologies. Section 2.2 describes how the SEAS ontologies are modularized and versioned. At last, Section 2.4 motivates the design choices for the ontologies.

2.1 General requirements

The ontologies and the server that exposes them on the Web need to satisfy the following best practices:

1. IRIs must satisfy the following best practices:
   (a) IRIs must not change, and must conform to the Semantic Web standards [2, 15];
   (b) The description of a concept SHOULD be accessible by looking up its IRI [1, item 3];
   (c) Different representations of this description should be served depending on who asks [2];
2. The ontologies are valid OWL ontologies, and satisfy the best practices for OWL documents as defined in [20, §3].
3. The ontologies satisfy the best practices for linked vocabularies [18];
4. The ontologies should reuse existing ontologies when appropriate.

Furthermore, we have design constraints for the SEAS ontologies:

4. The ontology must be divided in different modules that import each other, using the mechanism described in [20, §3.4];
5. Each module must be versioned, using the mechanism described in [20, §3.3];
6. All of the resources IRIs in the SEAS ontologies must be in the same namespace.
7. The whole SEAS ontology should be accessible from the IRI of the SEAS namespace.

Finally, the ontology server must pass the following tests:

8. One may access an ontology at its URL in different formats, depending on the HTTP Accept field:
   - text/html (in the browser) -> a HTML documentation of the ontology;
   - application/rdf+xml (in the ontology editor ”Protégé” for instance) -> the ontology as RDF+XML;
   - text/turtle -> the ontology as a Turtle document
9. Browsing to http://vowl.visualdataweb.org/webvowl/#iri=<<URLofanontology>> , one sees a nice graph that documents the ontology.

The sections devise solutions to satisfy all those requirements.

2.2 Modularization and versioning

The SEAS project envisions numerous use cases that need knowledge representation means for very different domains (e.g., Smart Homes, Micro Grids, Electric Vehicles, Electricity market, Distribution and Retail operators and clients, Weather Forecast). Not every part of the SEAS ontologies are needed by every implementation. Furthermore, we want to allow SEAS ontologies to be extended and updated incrementally, without breaking current implementations.

We hence modularize the SEAS ontology in a set of different ontology modules, each may have different versions. When a module "imports" other modules, it accepts and enriches their vocabularies and sets of logical axioms. The choice of modules is driven by a study of use cases and methodological principle.[1]

An ontology series O is identified by an IRI, and each of the ontology versions O_i are also identified by IRIs. Our requirements impose that: An ontology version O_i identified by IRI v_i in an ontology series O identified by IRI u SHOULD be accessible via the IRI v_i. Furthermore, if O_i is the latest version of the ontology series O, then it SHOULD also be accessible via IRI u.

An ontology document is a document, so one may return a HTTP code 200 OK. For instance, seas:BuildingOntology has two versions: seas:BuildingOntology-0.9 and seas:BuildingOntology-1.0; the latter being the latest version.

2.3 Metadata

Every SEAS ontology and every concept defined in those ontologies has a consistent set of associated metadata. Let be the following prefixes and namespaces:

```
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix dc: <http://purl.org/dc/terms/> .
@prefix vann: <http://purl.org/vocab/vann/> .
@prefix voaf: <http://purl.org/vocommons/voaf#> .
@prefix vs: <http://www.w3.org/2003/06/sw-vocab-status/ns#> .
@prefix cc: <http://creativecommons.org/ns#>.
@prefix foaf: <http://xmlns.com/foaf/0.1/>.
```

2.3.1 Ontology metadata.

The SEAS ontologies are modularized and versionned. Following the recommendations in [19], one hence add the following metadata to each ontology:

- type owl:Ontology;
- a owl:versionIRI;
- a owl:versionInfo;
- potentially a owl:priorVersion, that points to the URI of the previous module version;
- potentially one or more owl:imports that point to other ontologies (or ontology module) to import.

We also use recommended metadata for linked vocabularies [18].

- type voaf:Vocabulary;

[1] http://en.wikipedia.org/wiki/Ontology_modularization

- a dc:title, with a language tag;
- a dc:description, with a language tag;
- a dc:issued, which is a xsd:date;
- a dc:modified, which is a xsd:date;
- a dc:creator;
- zero or more dc:contributor;
- a cc:license: <https://www.apache.org/licenses/LICENSE-2.0>;
- a vann:preferredNamespacePrefix: "seas";
- a vann:preferredNamespaceUri: "https://w3id.org/seas/";

2.3.2 Concept metadata.

Following the recommended metadata for linked vocabularies [18], every concept is annotated at least with the following metadata:

- a rdfs:label, with a language tag;
- a rdfs:comment, with a language tag;
- a vs:term_status: one of "unstable", "testing", "stable", or "archaic";
- a rdfs:isDefinedBy, that points to the ontology that defines the concept.

2.4 Namespace and IRIs

Requirement 1.a imposes that resource IRIs are designed with simplicity, stability and manageability in mind. For stability and manageability, usually one uses redirection services such as http://purl.org/. Numerous ontologies have a purl.org namespace. The first SEAS ontologies used a purl.org namespace: http://purl.org/NET/seas#. However:

- purl.org is not an initiative of the W3C;
- the administration console of purl.org has been deactivated during the SEAS project, and one cannot manage redirections anymore.

We hence decided to use the recent initiative of the World Wide Web consortium, w3id.org. The goal of w3id.org is that: "All identifiers associated with the service are intended to be around for as long as the Web is around. This means decades, if not centeries."[2] The SEAS namespace hence uses the w3id.org host. We hence operated a pull request to project https://github.com/perma-id/w3id.org on GitHub, and now own a w3id PROJECT-ID called "seas". One may specify redirections to the actual SEAS servers that expose the knowledge model as ontologies. These redirections to actual SEAS servers are defined using Apache htaccess documents. For now, https://w3id.org/seas/ redirects to https://ci.emse.fr/seas/, which is hosted by ARMINES-Fayol. The sources of this website are open-source and reusable by third-parties, and available on the GitHub: https://github.com/thesmartenergy/seas. This allows anyone that knows GitHub to contribute to the SEAS ontologies.

[2] https://www.w3.org/community/perma-id/

Hence, the SEAS namespace is stable, and decoupled from the actual server that exposes the ontologies. In the future, https://w3id.org/seas/ may redirect to any another server, for instance at http://the-smart-energy.com. Before accepting a change in the redirection, w3id.org checks that the new target works as well as the previous target. (i.e., no 404 error on the home page for instance). Furthermore, the new target server should also be conformant with the specifications described in the rest of this section.

From requirement 1.a, resource IRIs must be either 303 IRIs or hash IRIs, because they are real-world objects. Suppose for instance that an ontology O with IRI OI defines the class of systems R.

Hash IRIs. Let RI#fragment be a hash IRI for R. Operating a HTTP GET to RI must return a document with HTTP code 200 (this satisfies 1.b). The returned document should define R. Hence OI must be equal to RI. For instance, RI = https://w3id.org/seas/system#System. Suppose. another resource R_2 in another ontology O_2 has a IRI OI2#fragment2. Requirement 6 imposes that string 'seas:' be the prefix for both namespaces OI# and OI2#. Which is impossible.

The solution is 303 IRIs. Let RI be a 303 IRI for R. Operating a HTTP GET to RI must return a HTTP code 303 See Also that redirects to a IRI RI2. The definition of R should be accessible via RI2.

Ontology O defines R. Hence we propose that RI redirects to OI. For instance, RI = https://w3id.org/seas/System redirects to OI = https://w3id.org/seas/SystemOntology. Hence Requirement 6 is satisfied and there exists a unique prefix for 'seas', whose extended version is:

@Prefix seas: <https://w3id.org/seas/>

This is registered at the well known service http://prefix.cc/seas. To sum up, when looking up a IRI RI:

- If RI identifies an ontology document O, then the ontology server returns O with HTTP code 200 OK;
- If RI identifies a resource R, then the ontology server redirects to the ontology O that defines R with HTTP code 303 See Also.

2.5 Reusing existing ontologies

The Semantic Web philosophy encourages the reuse of existing ontologies when appropriate. The SEAS Use Cases require models that are common to numerous other projects (e.g., Provenance, Time instants and intervals, Quantities and Units of Measure, Time, Sensor/Actuator description, Sensor Observations, Predictions), or that are not actual subdomains of the Energy domain (e.g., products and offers). As illustrated in deliverable D2.1.1, there exists some ontologies for some of these domains. It is worth reusing such existing ontologies instead of developing a whole new ontology that covers every domain. There are different ways to reuse existing ontologies, with different implications: imports, and alignments.

2.5.1 Direct imports

When dealing with OWL ontologies, the formal way to reuse an OWL ontology in another OWL ontology is to use the owl:imports ontology annotation. When an OWL processor encounters such an annotation, it attempts to access to the IRI of the imported ontology, and to load all entity declarations and all axioms of this ontology. This is further recursive.

Some existing ontologies have a great number of axioms, are inconsistent, or show a high computational complexity. These cases result in unpractical reasoning in the SEAS ontology. Hence, one must show extra care when choosing to import an existing OWL ontology. Other existing ontologies are not based on description logics, but instead define a set of inference and/or validation rules. This is the case

of the W3C Data Cube ontology for instance. Reusing such an ontology would mean that a processor conformant with the SEAS ontologies must be conformant with the existing ontology, hence implementing these rules. Again, this may lead to an increase in the complexity of reasoning or querying tasks.

When appropriate, existing ontologies are directly imported by some module in the SEAS ontologies, and can thus be considered as imported parts of the SEAS ontologies. For example, the OWL Time ontology is imported by the SEAS Time ontology. The GoodRelations ontology is imported by the SEAS Offering ontology.

2.5.2 Alignments

Alignments with existing ontologies are defined in external modules, that import both the SEAS ontologies and the existing ontology. This is the case for the W3C Semantic Sensor Network ontology and the Semantic Actuator Network ontology. Some users will not want to be forced to import the SSN ontology in order to use the Device ontology. We hence define the alignment in module seas:SSNAlignment, which imports seas:DeviceOntology and ssn:, and formalizes the alignment by means of properties:

- owl:sameAs (asserts that two IRIs identify the same individual)
- rdfs:subClassOf (C rdfs:subClassOf D means that all individuals in class C are also in class D)
- rdfs:subPropertyOf (see RDFS spec for more details)
- owl:equivalentClass, owl:equivalentProperty, etc.

One can choose between using only SEAS, or using SEAS in conjunction with SSN. Furthermore, using this alignment, a RDF graph that is described with SSN become compatible with SEAS.

Future work for 2017 include the design of other such modules to align SEAS with SAREF and ifcOWL.

2.5.3 Reusing existing ontologies in data.

Data conformant to the SEAS ontologies may also import other ontologies depending on the use case. For instance, the SEAS reference architecture implementation sometimes imports the basic geo, or the vcard ontology, to model geo-coordinates or addresses. The SEAS ontologies recommend the use of the following external ontologies:

- Goodrelations - http://www.heppnetz.de/ontologies/goodrelations/v1.html
- foaf – http://xmlns.com/foaf/spec/
- vcard for RDF - http://www.w3.org/TR/vcard-rdf/
- schema.org - http://schema.org/
- org - http://www.w3.org/TR/vocab-org/
- geo - http://www.w3.org/2003/01/geo/
- time - http://www.w3.org/2006/time#

Energy actors are instances of class gr:BusinessEntity. We encourage seas partners to expose a description of their company using these vocabularies. For example, the following snippet is a partial description of the CNR business entity, written using the turtle concrete RDF syntax.

```
@prefix cnr: <https://w3id.org/cnr#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix vcard: <http://www.w3.org/2006/vcard/ns#>.
@prefix skos: <http://www.w3.org/2004/02/skos/core#> .
@prefix seas: <https://w3id.org/seas/> .
@prefix gr: <http://purl.org/goodrelations/v1#> .
@prefix vcard: <http://www.w3.org/2006/vcard/ns#> .
@prefix geo: <http://www.w3.org/2003/01/geo/wgs84_pos#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix org: <http://www.w3.org/ns/org#> .

cnr:Compagnie%20Nationale%20du%20Rhône a gr:BusinessEntity ;
  rdfs:label "Compagnie Nationale du Rhône"@fr ;
  skos:prefLabel "Compagnie Nationale du Rhône"@fr ;
  vcard:organization-name "Compagnie Nationale du Rhône" ;
  skos:altLabel "CNR"@fr ;
  vcard:hasURL <http://www.cnr.tm.fr> ;
  org:hasUnit cnr:Siège%20Social%20de%20Lyon ;
  org:hasUnit cnr:Direction%20Port%20de%20Lyon ;
  org:hasUnit cnr:Direction%20Régionale%20de%20Belley ;
  org:hasUnit cnr:Direction%20Régionale%20de%20Vienne ;
  org:hasUnit cnr:Direction%20Régionale%20de%20Valence ;
  org:hasUnit cnr:Direction%20Régionale%20d%27Avignon .


cnr:Siège%20Social%20de%20Lyon a org:Organization ;
  skos:prefLabel "Siège Social de Lyon"@fr ;
  skos:altLabel "Siège Social" ;
  vcard:organization-name "Siège Social de Lyon" ;
  vcard:hasEmail <mailto:cnr.lyon@cnr.tm.fr> ;
  vcard:hasAddress [
    vcard:street-address "2, rue André Bonin" ;
    vcard:postal-code "69004" ;
    vcard:locality "Lyon" ;
    vcard:country-name "France" ] ;
  vcard:hasTelephone [
    a vcard:Voice ;
    vcard:hasValue "tel:+33472006969" ] ;
  vcard:hasTelephone [
    a vcard:Fax ;
    vcard:hasValue "tel:+33472106666" ] ;
  geo:location [ geo:lat 45.775005 ; geo:long 4.813461 ] ;
  vcard:hasGeo <geo:45.775005,4.813461> ;
  gr:hasPOS cnr:FR:CN1:P:00011 .
```

Class gr:BusinessEntity is the domain of the following properties, that may be used to represent the role of the business entity:

- gr:hasDUNS: a Dun & Bradstreet Data Universal Numbering System number
- gr:hasGlobalLocationNumber: the Global Location Number (sometimes also referred to as International Location Numbe)
- gr:hasISICv4: The International Standard of Industrial Classification of All Economic Activities, Revision 4 code. See http://unstats.un.org/unsd/cr/registry/isic-4.asp for more information.
- gr:hasNAICS: North American Industry Classification System code. See http://www.census.gov/eos/www/naics/ for more details.

2.6 Solution implemented in the ontologies website

The ontologies are hence exposed in conformance with the specifications described in the previous sections, the linked data and the linked vocabularies best practices. Hence:

1. they are available on the Web with an open licence Apache 2.0;
2. they are available in a structured format processable by machines (e.g., Excel instead of an image);
3. they are available in a non-proprietary format (e.g., CSV instead of Excel);
4. they are described using W3C standards (3 first principles of the linked data);
5. they are linked to other datasets, using RDF links (linked data 4th principle);
6. they are described with provenance metadata, licence, etc.

They furthermore satisfy the following constraints:

- the SEAS ontologies are made of different modules, each module being versioned, in conformance with [20, §3.1 to 3.4];
- an HTTP GET to a concept's URI 303 redirects to the most recent version of the module where this concept is defined;
- each module version is available in the Turtle, RDF/XML, and HTML formats, with server content negociation, reference to a unique canonical URI for each representation, and hint for a filename to use if the browser wants to download the file. For example, operating a HTTP GET to https://w3id.org/seas/EvaluationOntology with Accept: text/turtle, one gets an HTTP response with the following headers:

  Content-Disposition: filename= EvaluationOntology-1.0.ttl;
  Content-Location: https://w3id.org/seas/EvaluationOntology-1.0.ttl

- every concept's URI in every version of every module uses a single namespace https://w3id.org/seas/ abbreviated by seas:

One may check that all the requirements are met, and all the tests pass. Hence, all the ontologies, including the main ontology, and all the resources, share the same namespace:

@prefix seas: <https://w3id.org/seas/>

In order for the SEAS ontologies to be better referenced, and in order for some well-known RDF and SPARQL editors with autocompletion to be able to discover them:

- The namespace seas: has been registered on the website http://prefix.cc;
- The ontology has been registered on the Linked Open Vocabularies (http://lov.okfn.org/dataset/lov/vocabs/seas).

The source of the website that exposes the SEAS ontologies is open-source and available at https://github.com/thesmartenergy/seas. It uses an open-source tool developed by ARMINES-Fayol and reusable by third-parties that checks the quality of the ontologies, and exposes them on the web. This tool is available and documented at: https://w3id.org/ontop/.

2.6.1 Accessing the main ontology

The main ontology that glues all of the module ontologies has IRI https://w3id.org/seas/.

A HTTP GET operation to https://w3id.org/seas/ returns a 302 Found redirection (temporary redirection) to http://ci.emse.fr/seas/

A HTTP GET operation to http://ci.emse.fr/seas/ returns a 202 OK, and the actual returned representation depends on the Accept header:

- If it contains text/turtle, one retrieves the turtle document.
- If it contains application/rdf+xml, one retrieves the RDF/XML document.
- Else, one retrieves the HTML document that describes the ontology and the ontology server implementation (this is what one sees when one accesses to this URL in a browser).

This ontology has version 1.0, and version IRI https://w3id.org/seas/seas-1.0. A HTTP GET operation to https://w3id.org/seas/seas-1.0 returns a 302 Found redirection (temporary redirection) to http://ci.emse.fr/seas/seas-1.0 . A HTTP GET operation to http://ci.emse.fr/seas/ returns a 202 OK, and the actual returned representation depends on the Accept header:

- If it contains text/turtle, one retrieve the turtle document;
- If it contains application/rdf+xml, one retrieve the RDF/XML document.

2.6.2 Accessing a module ontology

The main ontology imports all other module ontologies. In the Protégé editor, loading this ontology from its IRI or its version IRI works well, and all the imported ontologies are also loaded. Doing so is the easiest way to get statistics about the SEAS ontologies. A HTTP GET operation to https://w3id.org/seas/BuildingOntology returns a 302 Found redirection (temporary redirection) to http://ci.emse.fr/seas/BuildingOntology. A HTTP GET operation to http://ci.emse.fr/seas/BuildingOntology returns a 202 OK, and the actual returned representation depends on the Accept header:

- If it contains text/turtle, one retrieve the turtle document.
- If it contains application/rdf+xml, one retrieve the RDF/XML document.
- Else, one retrieve the human-readable documentation in HTML.

2.6.3 Accessing a resource

The module ontology https://w3id.org/seas/BuildingOntology defines for instance resource seas:Building, which expands to IRI https://w3id.org/seas/Building. A HTTP GET operation to https://w3id.org/seas/Building returns a 302 Found redirection (temporary redirection) to http://ci.emse.fr/seas/Building. A HTTP GET operation to http://ci.emse.fr/seas/Building returns a 303 See Also to http://ci.emse.fr/seas/BuildingOntology, i.e., the ontology that defines it. In the Protégé editor, trying to load an ontology from IRI http://ci.emse.fr/seas/Building actually loads the ontology http://ci.emse.fr/seas/BuildingOntology.

3 Knowledge engineering methodologies

T2.2 participants used a combination of bottom-up and top-down approaches to develop the SEAS ontologies. In the top-down approach, knowledge engineers propose a model, and then see how it can represent the data of the SEAS partners. The bottom-up approach consists in asking the SEAS partners to collaboratively contribute to the SEAS ontologies, for instance by listing concepts that they need to represent. This section overviews the approaches undertaken.

3.1 Traditional knowledge engineering approach (ARMINES-Fayol)

The traditional knowledge engineering approach consists in studying texts of the domain, interviewing domain experts, and developing the knowledge model to satisfy their needs. As SEAS has numerous partners around Europe, this approach could not scale. Nevertheless, this approach has been led with:

- An extensive description of some use cases. This is an output of WP1. Use cases 2 and 3 in Table 1 have been mainly used.
- Interviews with partners such as GECAD, CNR, Ubiant, or ENGIE as a first step based on concrete cases.

UC Number	UC Name	Author
001	Energy dashboard for end users	SIVECO-ECRO
002	Public EV Charging Station with stationary battery load control	CNR
003	EVSE with Demand Side Management	CNR
005	Decentralized prosumers energy market	ARMINES-FAYOL
007	Day-ahead microgrid resources management considering the impact of previewed consumption and market prices	GECAD
009	Intelligent Building Automation	SPAIN TECH
012	A stochastic approach for smart home energy management	INNOVA
013	Carbon free EV	ICAM
017	Load curve classification	GECAD
018	Demonstrator ENORD	ISA
019	Smart building management	EVOLEO
032-033-035	Customer awareness with daily load curve	BASKENT
046	Consumer wants to influence on other users	ASEMA
052	Direct load control command	EKE
053	Interactive production-consumption forecast and control	EKE
065	Distributed resource management for controlling the local micro-generation peaks	EMPOWER
075-076	Online algorithm for minimizing operating costs of energy systems in buildings and reducing electricity peak demand in general (demand-side management)	FORTUM
082	µ-CHP+HEMS+battery for B2C market	GDF SUEZ

Table 1: List of reviewed Use Case (table taken from deliverable D1.3: Communication Architecture)

3.2 UML-based development and importing-exporting tools (VTT)

Part of the initial SEAS information modeling task was performed by analysis of activities and information related to use cases and the analysis work was performed using UML tool. The result of this analysis is a draft of SEAS ontologies in UML format. The benefit of using of UML tool for this task is that it provides means to manage large information models, support for easily modularising the model into packages while providing several views to the same underlying model independently of the package structure.

The results of UML based information modeling task are presented in SEAS D2.1 [4]. Figure 2 shows an initial view of how that work was related to the ontology/knowledge modeling presented in the previous version of this deliverable. The UML modeling process is described in more detail in [4].

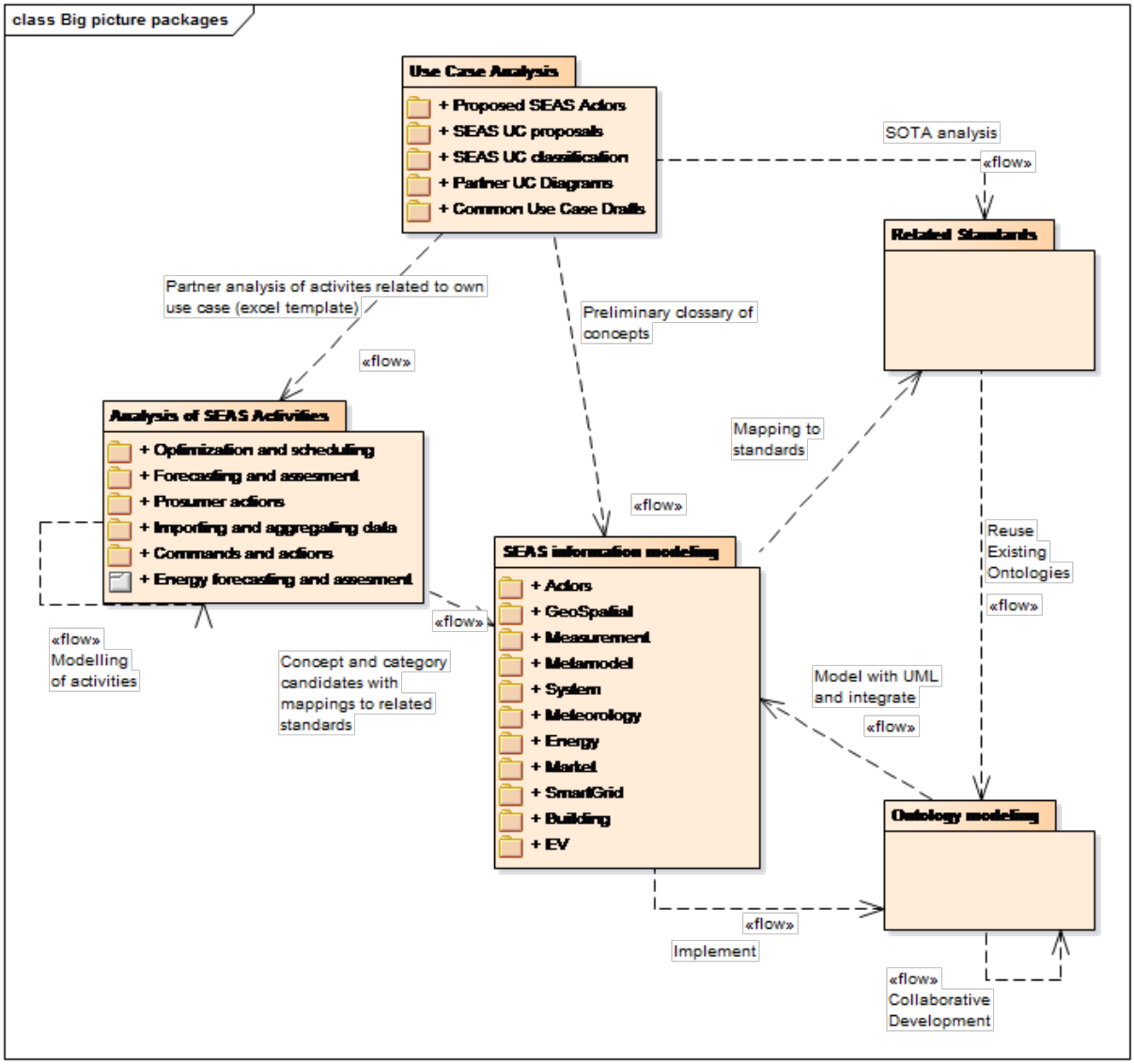

Figure 2: Relation of SEAS UML modeling to SEAS Ontology modeling

The UML model is available as a read-only website exported from Enterprise Architect UML tool. The packages of UML information model typically map directly to SEAS ontology modules but typically the analysis scope of UML model is more general and all of the concepts and associations presented there are not implemented in SEAS ontologies.

UML class modeling language semantics differs from the semantics of modeling languages like RDFs and OWL in several aspects and complete transformation between those is probably not possible but both informative or tool supported partial mappings have been proposed [21]. Several UML tools like Protégé import and export mechanism between UML and semantic models.

There are several model driven ontology development approaches that were evaluated for feasibility for helping to develop and maintain SEAS ontologies:

- Reusing existing UML based information models as directly as SEAS information model and transforming them to knowledge model implementation.
- Reusing existing UML profiles for information models such as Common Information Model [3].
- modeling ontologies with OWL based UML profile for Ontology Definition Metamodel [17].

Direct reuse of existing standards based models and profiles was considered problematic because of their typically quite large size and licensing restrictions. UML profile for ODM was found useful for importing and visualizing existing ontologies (also quite large) ontologies. Especially the ability of UML tool to provide several views to the ontology was considered useful while the readability of models as opposed to traditional UML was considered worse because also the properties are presented as class symbols.

The poor maturity of the tool plug-in caused problems that made this approach infeasible for implementing the knowledge model. The existing ontologies had to be edited to be able to import into tool and parts of ontology were not imported at all. Exporting new UML RDF models back to RDF format did not work in latest version of EA tool. With better maturity the profile could be useful as a intermediate representation between traditional UML based version of knowledge model and RDFs/OWL implementation of model.

A simplified UML modeling approach was adopted that benefits from syntactic compactness of UML class model and support for package namespaces. Typical mapping of UML to OWL is represented in Table 2.

UML	RDFs/OWL	Note
Package	owl:Ontology	UML does not define unique IRI
Class	owl:Class with same name	Class concept of sematic languages is related to concept of set
Subclass relation	rdfs:subClassOf	
Attribute	owl:DataypeProperty or owl:ObjectProperty with same name	Simple datatypes can be mapped to datatype properties.
Binary association	owl:ObjectProperty with same name	UML has no concept of subProperties
Aggregation association	has(Member)/isMemberOf owl:ObjectProperty	Aggregation is nontransitive relation with inverse relation
Composition association	hasPart/isPartOf owl:ObjectProperty	Composition is transitive and inverse functional relation.
UML actor	owl:Class with same name	Actor is typically a Concrete SEAS Entity
UML business actor	owl:Class with same name	Business actor is typically a (Abstract) Role taken by a SEAS Entity
UML Activity	owl:Class	SEAS Activity (Abstract Entity)

Table 2: Relation between UML and the version 0.9 of the SEAS ontologies.

UML has a shorthand for showing subclass relation in right hand corner of the class symbol when the parent class is not in same diagram. This is especially useful when showing compact views of ontology.

UML syntax and tools allow using UML classes as namespaces i.e. contain classes inside the parent class. The resulting qualified namespace can be shown on top of class symbol. This is sometimes used in SEAS UML model drafts as a compact way to define a set of related subclasses without need to show the often numerous subclass associations to the parent classes in same diagram but this is against semantics of UML and usually this is later modified to use separate views instead.

3.3 Protégé (IMT, Télécom Sud-Paris)

The latest version of Protégé has also been used to formalize part of the SEAS semantic open data model and ontology for the following reasons:

- It fully supports the latest OWL 2 Web Ontology Language and RDF specifications from the World Wide Web Consortium,
- Mostly used,
- Provides graphic user interface,
- Can import existing ontologies,
- Extensible, and provides a plug-and-play environment,
- Clear visualization of the classes and properties (OntoGraph plug-in, OWLViz),
- Support wide range of reasoners (e.g. Pellet reasoner),

3.4 SEAS Wiki (ARMINES-Fayol)

As Knowledge engineers, T 2.2 participants usually need to directly interact with domain experts to develop a Knowledge Model. Yet these domain experts are distributed around Europe, and are not Semantic Web experts. Hence we wanted a tool to enable the collaborative development of the Knowledge Model, by domain experts. This led ARMINES-Fayol to focus during the first semester of 2015 on the development of LinkedVocabularyEditor, a MediaWiki extension that transforms a MediaWiki instance into (i) a collaborative ontology editor by the domain experts, and (ii) a publication platform conformant with the Linked Vocabulary best practices. This MediaWiki extension has been the object of a demonstration paper at IC 2015 conference.

Identified needs. The goal of LinkedVocabularyEditor is to answer the following needs:

- The tool must offer simplified visualization and edition interfaces for an ontology and its different modules;
- The tool must enable knowledge experts to control the ontology evolution: enable change notifications and version control;
- The tool must serve as an ontology publication platform, conformant with the best practices stated by the W3C;
- The tool must be integrated in a common open-source collaborative work tool, in order to encourage a community of developers to enhance it.

Existing tools do not answer all of these needs. For instance, Protégé and its online version (Noy, 2001, Tudorache, 2008) target knowledge engineers. Neologism platform (Basca, 2008) do not enable to add complex axioms to the ontology, and do not enable version control. Semantic MediaWiki (Krötzsch, 2006) and WikiData (Vrandecic, 2012) focus respectively on annotating links in wiki pages, and on structured data edition, rather than on the edition of vocabularies that structure these annotations and data.

Basics of the extension. The LinkedVocabularyEditor extension defines a wiki namespace called Resource:, synchronized with a ARC2 triple store. Every page in this wiki namespace represents a resource as follows: If prefix seas: represents https://w3id.org/seas# (which is not true outside of this section), then page Resource:Seas:ElectricalSystem describes resource https://w3id.org/seas#ElectricalSystem. When one accesses this page, a content negotiation process is led, and the wiki serves a HTML, RDF/XML, turtle, or N-triples depending on the request's Accept HTTP header option.

Internally, the content of a page is a JSON-LD representation of the described resource, in every named graph of the ARC2 triple store this resource belongs to. The content of the HTML page is generated not only from this JSON-LD, but also thanks to calls to the ARC2 SPARQL endpoint. The edition interface is dynamically built using angularjs, and calls to the ARC2 SPARQL endpoint to look up resources in the wiki.

The use of the purl.org redirection medium, with hash-based naming convention is fully compatible with the MediaWiki extension.

Use. When a domain expert creates a new page or edits an existing page, a simple form is served, and the submission of this form updates the ARC2 triple store. The extension already enables the edition of ontologies with OWL LD profile (Glimm, 2012). The Knowledge Engineer can use already implemented MediaWiki functionalities for being notified, moderating, and controlling versions. Moreover, it has access to two special wiki pages. One enables him to edit RDF namespaces (to define a namespace for a new ontology module for instance), and another enables him to export and import existing ontologies. This last special page enables him to export the ontology, manually edit it, and import it again in the wiki.

Current limitations, and possible evolutions. LinkedVocabularyEditor implements a small set of essential functionalites, that form an interesting basis for the development and the evaluation of new ontology visualization and collaborative ontology edition techniques. Other functionalities are missing, as the renaming or the deletion of resources. Furthermore, some MediaWiki functionalities would be worth reused, as the categories hierarchy to represent hierarchies of classes or properties. Finally, every page contains only those triples that have the described resource as subject. This temporary choice avoids inter-dependencies between pages content, and non-trivial research and engineering questions will be raised when we will suppress this limitation.

3.5 The SEAS Workshop (ARMINES-Fayol and ENGIE)

We organized a Knowledge Engineering workshop in December 2015. The goals of this workshop was to:

- Familiarize the participants with the Semantic Web formalisms and technologies;
- Network with other related projects (eMI3, SWIMing Project) and potential SEAS clubbers;
- Capture requirements for a few well identified topics;
- Boost the development of the SEAS ontologies;
- Help partners in charge of pilots to start working on their implementations.

The workshop gathered 45 participants over 3 days, had 6 keynote speakers, and 6 sessions of parallel knowledge capture sessions, each animated by a knowledge expert, during which a specific aspect of the SEAS ontologies has been discussed with the domain experts.

The website of the workshop is available at:

http://data.the-smart-energy.com/workshop/2015/12/report.html

Videos of the Workshop. videolectures.net accepted to create a dedicated section for the SEAS project in the EU Supported section. The videos of this workshops are available at the following URL:

https://videolectures.net/SEASworkshop2015_paris/

Exploitation of the Workshop. The workshop confirmed that the SEAS wiki should be abandoned, and that VoCamps are a great mean to get domain experts involved in the development of the Knowledge Model. The T2.2 participants exploited the results of the Workshop sessions during 2016.

4 Description of the SEAS ontologies (ARMINES-Fayol)

The SEAS ontologies design took as input the results of work packages WP 1: Use Cases and Business Models, and WP 2.1: SEAS Information Analysis –see deliverable D2.1.1 [4]. Apart from introducing Knowledge Representation and the Semantic Web formalisms in Section 4.1, deliverable D2.1.1 reports on an analysis of existing work related to the two first levels of understanding: data and information, depicted in Figure 1. Furthermore, deliverable D2.1.1 lists and describes domains that potentially need to be modeled, along with existing ontologies for these domains. Finally, D2.1.1 describes a meta-model that has been proposed from a first Use Cases analysis to support the SEAS ontologies development.

This section first gives an overview of the SEAS ontologies in Section 4.1. The core modules, described in Section 4.2, define some basic design patterns that are instantiated in other modules. Section 4.3 describes modules that instantiate the Process design pattern. Then, Section 4.4 describes modules that define properties and evaluations. Finally, vertical modules for the Smart Grid, Smart Home, the SEAS architecture, and Offers and markets, are described in sections 4.5, 4.6, 4.7 and 4.8, respectively.

4.1 Overview

The SEAS knowledge model is a modularized and versioned ontology that describes among other energy systems and their interactions. The core of the ontology is made of four modules:

- seas:FeatureOfInterestOntology – this modules defines a design pattern to describe features of interest (e.g., a room) and their properties (e.g., its temperature) that are qualifiable, quantifiable, observable, or actionnable.
- seas:EvaluationOntology – this module defines design patterns to describe evaluations (qualification or quantification) of properties, and to qualify these evaluations: (i) what kind of evaluation: (e.g., maximum tolerable temperature, forecasted temperature, mean temperature), (ii) in a spatio-temporal validity context (e.g., temperature of the room between 10 am and 12 am tomorrow).
- seas:SystemOntology – this module defines a design pattern where systems are connected via connection points. This design pattern can be instantiated for example to describe zones inside a building (systems), that share a frontier (connections). The properties of a system are typically state variables (e.g., agents population, temperature), the properties of connections are typically flows (e.g., heat flow).

- Process Execution Platform ontology, an externalized module that generalizes the SSN ontology and describes pep:ProcessExecutors (sensor, actuator, web service, etc.) that implement pep:Processes and execute pep:ProcessExecutions.

On top of these core modules, the SEAS ontology defines several "vertical" modules, that are dependent of a domain. For example:

- seas:ElectricPowerSystemOntology – this module defines system types (e.g., seas:ElectricPowerProducer, seas:ElectricPowerStorageSystem), their connections through which they exchange energy (e.g., seas:ThreePhasePowerBus), their properties (e.g., seas:power, seas:voltage), and the classical types of evaluation for these properties (e.g., seas:ActiveEvaluation, seas:ReactiveEvaluation, seas:NominalOperatingEvaluation), the common sensors and actuators.
- seas:ZoneOntology – this module defines physical zones and their frontiers, through which these zones can exchange agents, thermic energy, light, or humidity.

Other vertical modules are proposed for energy markets, demand/response, etc. Labels and comments for the concepts are available at least in english, but sometimes in french, finnish, and/or portugese. The ontologies use concepts from external vocabularies:

- for the description metadata: rdfs, owl, cc, vann, voaf, foaf, dc;
- reuse of: time, geo, pep, gr;
- formal alignments defined in separate modules to: ssn, qudt.

Table 3 lists some of the metrics one obtains when loading the SEAS ontologies in the Protégé editor. The total number of concepts (classes + object properties + data properties + individuals) is currently of 610. These ontologies are published in Turtle, XML, and HTML according to the Web of linked vocabularies best practices. Their documentation is automatically generated in HTML thanks to the ONTOP tool (ARMINES-Fayol).[3]. The source of the website that exposes these ontologies is open source and available at https://github.com/thesmartenergy/seas.

	PEP	SEAS
Axioms	92	4174
Logical axioms	27	1274
Classes	7	293
Object properties	10	214
Data properties	-	6
Individuals	1	44

Table 3: Number of entities in the PEP and the SEAS ontologies.

The modules and how they import each other is illustrated on figure 3.

4.2 Core modules

4.2.1 Core module FeatureOfInterest

IRI: https://w3id.org/seas/FeatureOfInterestOntology
Current version IRI: https://w3id.org/seas/FeatureOfInterestOntology-1.0

[3]ONTOP – https://w3id.org/ontop/

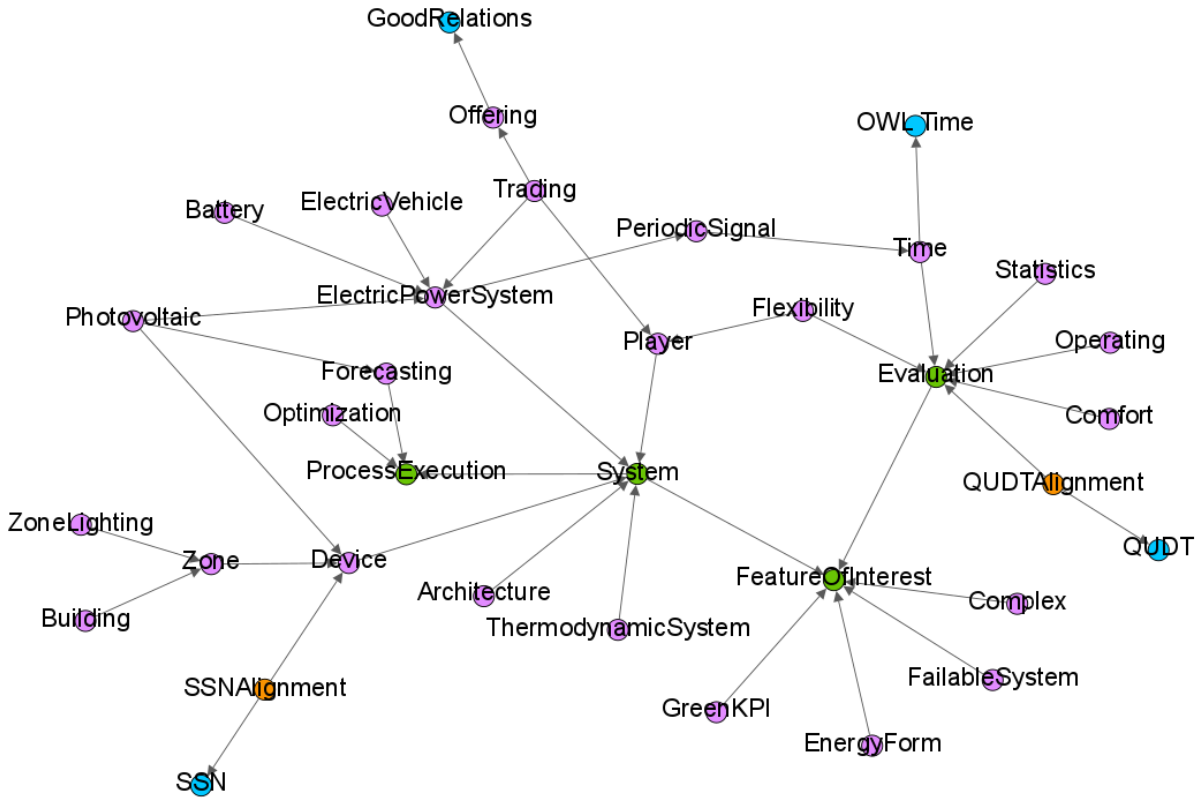

Figure 3: Illustration of the modules and their imports. Green nodes are core modules, pink nodes are vertical modules, orange nodes are alignment modules, and blue nodes are external modules.

seas:FeatureOfInterestOntology is the only SEAS module that imports no other module. This ontology defines features of interest, and their properties.

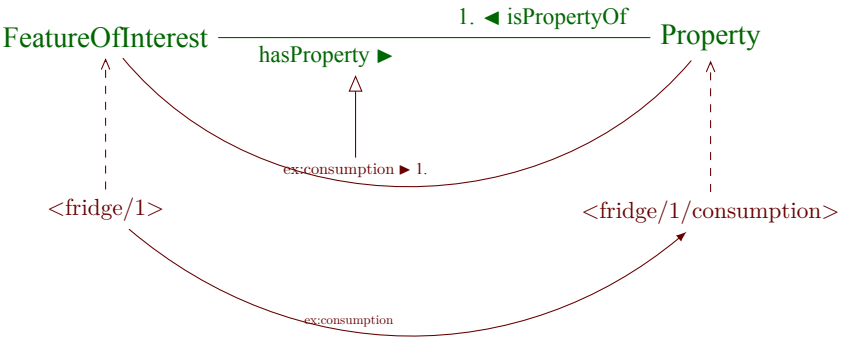

Figure 4: Illustration of module seas:FeatureOfInterestOntology and its use.

High level description. A feature of interest is an abstraction of a real world phenomena (thing, person, event, etc.). A feature of interest is then defined in terms of its properties, which are qualifiable, quantifiable, observable or operable qualities of the feature of interest. For example the following RDF Graph describes a fridge, with its property <fridge/1/consumption>:

```
<fridge/1> a seas:FeatureOfInterest ;
  seas:hasProperty <fridge/1/consumption> .
```

Design pattern: new properties One can refine the relationship between <fridge/1> and <fridge/1/consumption>. For example, a fridge has exactly one energy consumption.

The design pattern to define an actual property relation is to define it as a functional sub properties of seas:hasProperty, with domain seas:FeatureOfInterest and range seas:Property. For example,

```
ex:consumption a owl:ObjectProperty , owl:FunctionalProperty ;
  rdfs:subPropertyOf seas:hasProperty ;
  rdfs:domain seas:FeatureOfInterest ;
  rdfs:range seas:Property .
```

Then the link between <fridge/1> and property <fridge/1/consumption> can be further specified:

```
<fridge/1> a seas:FeatureOfInterest ;
  <consumption> <fridge/1/consumption> .
```

Module seas:BooleanPropertyOntology illustrates the application of this design pattern for two properties: seas:onStatus and seas:openStatus.

Design pattern: property sub-classes One can define a hierarchy of property subclasses, which can then be used for classifying properties with respect to the way they can be evaluated. Module seas:BooleanPropertyOntology illustrates the application of this design pattern for two sub-classes of seas:Property:

- seas:EnumeratedProperty: A property that can have one of multiple enumerated values;
- seas:BooleanProperty: A property that can be true of false. May be used to model commands like on/off.

Every ontology can define other property sub-classes such as seas:LengthProperty, seas:PowerProperty, seas:EnergyProperty, and specify that evaluations of such properties all have the same dimension.

```
seas:LengthProperty a owl:Class ;
  rdfs:subClassOf seas:Property ;
  rdfs:subClassOf [
      a owl:Restriction ;
      owl:onProperty seas:value ;
      owl:allValuesFrom cdt:length ] .
```

The definition of seas:value is described in Section 4.2.2.

Differences with the former SSN ontology. Unlike in the 2005 SSN ontology, any feature of interest property may itself be a feature of interest. In fact, the SSN ontology was originally based on the DUL ontology, which imposed that properties could not be feature of interest themselves. Yet we identified in some use cases where properties should have properties.

For example, the fridge consumption has a frequency, or a faults counter.

```
<fridge/1/consumption> a seas:FeatureOfInterest ;
  ex:frequency <fridge/1/consumption/frequency> .
```

One may also define the class of electric power systems as the class of feature of interest that are linked to some property by property key <consumption>:

```
<ElectricPowerSystem> a owl:Class ;
  rdfs:subClassOf [
    owl:onProperty <consumption> ;
    owl:someValuesFrom seas:EnergyProperty ] .
```

Alignment with the new SSN. The ontology seas:FeatureOfInterestOntology is designed to be aligned with the core of the SSN ontology that is currently being standardized in the W3C "Spatial Data on the Web" working group. Another module describes these alignments:

IRI:	https://w3id.org/seas/SSNAlignment
Current version IRI:	https://w3id.org/seas/SSNAlignment-1.0

4.2.2 Core module EvaluationOntology

IRI:	https://w3id.org/seas/EvaluationOntology
Current version IRI:	https://w3id.org/seas/EvaluationOntology-1.0

seas:EvaluationOntology imports only seas:FeatureOfInterestOntology. It enables to describe evaluations for properties of features of interest.

Direct evaluation A seas:Property may be directly associated with a quality or quantity value, which is then unique and constant. This association is asserted using property seas:value.

A quantity value may use external vocabularies such as QUDT (it would then be qudt:Quantity), or OM (it would then be a om:Quantity), or be directly encoded as a literal using an appropriate datatype. We encourage the use of the cdt:ucum custom datatype. The lexical value of a cdt:ucum is defined as follows:

```
unum ::= number ' ' unit
number ::= noDecimalPtNumeral | decimalPtNumeral | scientificNotationNumeral
unit ::= (see the Unified Code of Units of Measure specification)
```

noDecimalPtNumeral, decimalPtNumeral and scientificNotationNumeral are defined in the XSD 1.1 recommendation. This datatype must either be hardcoded in RDF engines, or could be specified as a Linked Datatype, making it easy for any SPARQL engine to support the comparison of quantity values. Linked Datatypes are described in [8]. For example, the following triples quantify the consumption of a fridge using cdt:ucum literals:

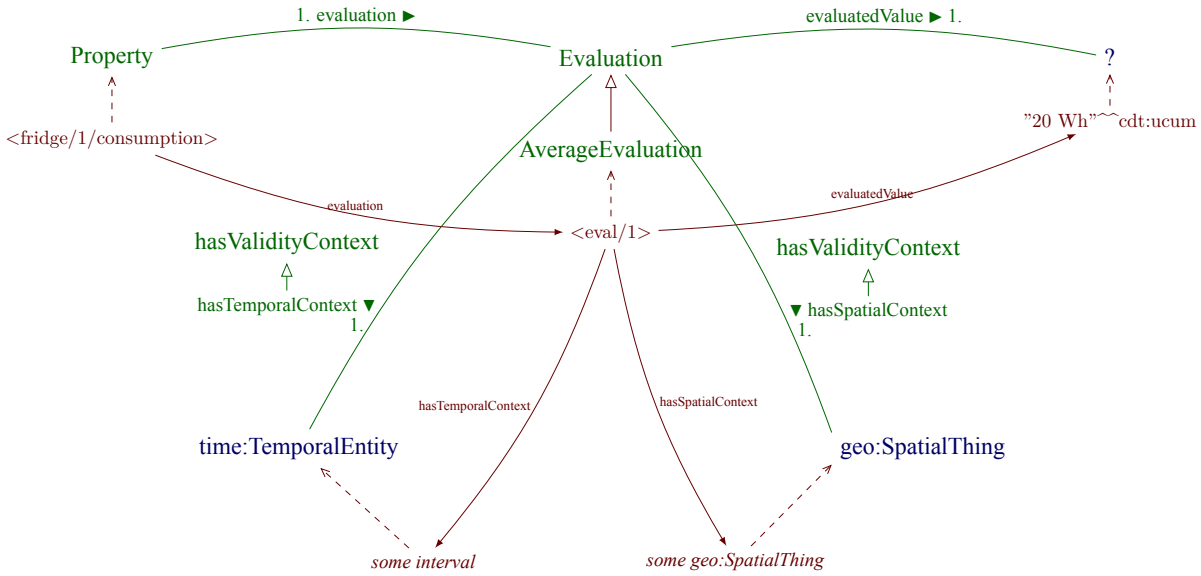

Figure 5: Illustration of module seas:EvaluationOntology and its use.

```
<fridge/1/consumption/frequency> seas:value "50.1 Hz"^^cdt:ucum .
<fridge/1/consumption/voltage> seas:value "231 V"^^cdt:ucum .
<fridge/1/consumption/tension> seas:value "2432 mA"^^cdt:ucum .
<fridge/1/consumption/voltageTensionPhase> seas:value "1.68 RAD"^^cdt:ucum .
```

Quantified evaluations. Because property values may evolve in space and time, or because they can be approximate measures or forecasts, class seas:Evaluation qualifies the link seas:value. In particular, an instance of seas:Evaluation may hold metadata about:

1. **The type of evaluation** is defined by the hierarchy of seas:Evaluation sub classes. This hierarchy includes classes such as:

   - seas:TimeAverageEvaluation: the given value is the average of the property value over its temporal context;
   - seas:MaximumOperatingEvaluation: the given value is the maximum operating value for the property in all of its validity context.

2. **The context of validity of the evaluation.** The W3C Spatial Data on the Web Working Group may define a best practice for describing the validity context of some entity. See Spatial Data on the Web Best Practices, W3C First Public Working Draft 19 January 2016:

   > Best Practice 11: How to describe properties that change over time [...] When entities and their properties can change over time, or are valid only at a given time, and this needs to be captured, it is important to specify a clear relationship between property data and its versioning information.

   As soon as this is the case, we recommend to use the proposed best practice. In the meantime, an evaluation validity context is described using functional sub properties of seas:hasValidityContext. The SEAS ontologies define two such properties as shown in Figure 5:

   - seas:hasTemporalContext links an entity to its temporal validity context, a time:TemporalEntity;
   - seas:hasSpatialContext links an entity to its spatial validity context, a geo:SpatialThing.

3. **Provenance information or any other data.** Other metadata may be added to describe an evaluation instance. For example the W3C PROV Ontology enables to describe the activity that generated the evaluation, or its generation time. Other vocabularies may be used to further describe evaluations. See the Linked Open Vocabulary cloud.

Example 1. *The day-ahead forecasted temperature at Aéroport de Lyon is 28.3 °C:*

```
<air/temperature> seas:evaluation [
  a seas:TemperatureEvaluation , seas:exactEvaluation , seas:Forecast ;
  prov:wasGeneratedBy <algorithm/1/execution/234> ;
  prov:generatedAtTime "2016-08-12T12:00:00Z"^^xsd:dateTime ;
  seas:hasTemporalContext [ a time:Instant ; time:inXSDDateTime "2016-08-13T12:00:00Z"^^xsd:dateTime ] ;
  seas:hasSpatialContext [ a geo:Point ; geo:lat 45.7242502 ; geo:long 5.0914517 ] ;
  seas:evaluatedValue "28.3 °C"^^cdt:ucum ] .
```

Example 2. *The average frequency of property* <fridge/1/consumption/frequency> *over time interval 12:00 - 13:00, as generated by* <algorithm/1/execution/12>*, is 50.1054 Hz:*

```
<fridge/1/consumption/frequency> seas:evaluation [
  a seas:FrequencyEvaluation , seas:TimeAverageEvaluation ;
  prov:wasGeneratedBy <algorithm/2/execution/12> ;
  seas:hasTemporalContext [ a time:Interval ; time:hasBeginning [ time:inXSDDateTime
      "2016-09-10T12:00:00Z"^^xsd:dateTime ] ; time:hasEnd [ time:inXSDDateTime
      "2016-09-10T13:00:00Z"^^xsd:dateTime ] ] ;
  seas:evaluatedValue "50.1054 Hz"^^cdt:ucum ] .
```

Example 3. *The minimal operating value of* <fridge/1/consumption/frequency> *is 47.0 Hz:*

```
<fridge/1/consumption/frequency> seas:evaluation [
  a seas:MinimalOperatingValueEvaluation ;
  seas:evaluatedValue "47.0 Hz"^^cdt:ucum ] .
```

Alignment with QUDT. A seas:Property may be quantifiable. Hence qudt:Quantity can be considered as a sub-class of seas:Property, and seas:Evaluation can be considered equivalent to qudt:QuantityValue. The alignment with QUDT is defined in a separate module, that imports seas:EvaluationOntology:

IRI:	https://w3id.org/seas/QUDTAlignment
Current version IRI:	https://w3id.org/seas/QUDTAlignment-1.0

Choice criteria When properties seem similar and inter-dependent, one requires precise criteria to decide how the represent them best.

Proposition a: distinct properties. If properties can be independent, one represent each of them distinctly, as functional sub-properties of property seas:hasProperty. This is the chosen model to represent the length and the width for example:

```
ex:length a owl:ObjectProperty , owl:FunctionalProperty ;
  rdfs:subPropertyOf seas:hasProperty ;
  rdfs:range seas:LengthProperty .

ex:width a owl:ObjectProperty , owl:FunctionalProperty ;
  rdfs:subPropertyOf seas:hasProperty ;
  rdfs:range seas:WidthProperty .
```

One can hence give independent evaluation for each of them.

Proposition b: a property and different evaluation sub-classes. If the property is unique, but one qualify/quantity it differently, then one only define a single functional sub-property of property seas:hasProperty, and several sub-classes of seas:Evaluation. This is the chosen model to represent a length, and the evaluation of its minimum and maximum:

```
seas:length a owl:ObjectProperty , owl:FunctionalProperty ;
  rdfs:subPropertyOf seas:hasProperty ;
  rdfs:range seas:LengthProperty .

seas:MinimumEvaluation a owl:Class ;
  rdfs:subClassOf seas:Evaluation .

seas:MaximumEvaluation a owl:Class ;
  rdfs:subClassOf seas:Evaluation .
```

Note: the physical dimension given to a property must be the same as the physical dimension of each of its evaluation. The following snippet models the active, reactive, and power factor of a system consumption:

```
seas:activeEnergy a owl:ObjectProperty , owl:FunctionalProperty ;
  rdfs:subPropertyOf seas:hasProperty .

seas:reactiveEnergy a owl:ObjectProperty , owl:FunctionalProperty ;
  rdfs:subPropertyOf seas:hasProperty .

seas:powerFactor a owl:ObjectProperty , owl:FunctionalProperty ;
  rdfs:subPropertyOf seas:hasProperty .
```

Proposition c: a property and several properties of this property. If one wants to model a unique property that itself has several properties with different quantity dimensions. This is the case for the consumed energy, with its active, reactive, apparent, and the power factor:

```
seas:energy a owl:ObjectProperty , owl:FunctionalProperty ;
  rdfs:subPropertyOf seas:hasProperty ;
  rdfs:range seas:EnergyProperty .

seas:EnergyProperty a owl:Class ;
  rdfs:subClassOf seas:Property .

seas:reactiveEnergy a owl:ObjectProperty , owl:FunctionalProperty ;
  rdfs:subPropertyOf seas:hasProperty ;
  rdfs:domain seas:EnergyProperty .

seas:powerFactor a owl:ObjectProperty , owl:FunctionalProperty ;
  rdfs:subPropertyOf seas:hasProperty ;
  rdfs:domain seas:EnergyProperty .
```

Using OWL axioms property (or rules), one may express an equivalence between propositions (a) and (c) above.

4.2.3 Core module SystemOntology

IRI: https://w3id.org/seas/SystemOntology
Current version IRI: https://w3id.org/seas/SystemOntology-1.0

seas:SystemOntology only imports seas:FeatureOfInterestOntology. This module defines a design pattern where systems are connected via connection points.

High level description

A system, modeled by class seas:System, is part of the univers that is virtually isolated from the environment. The system properties are typically state variables (e.g., consumed or stored energy, agent population, temperature, volume, humidity). Figure 6 illustrates classes and properties that can be used to define connected systems and their sub-systems.

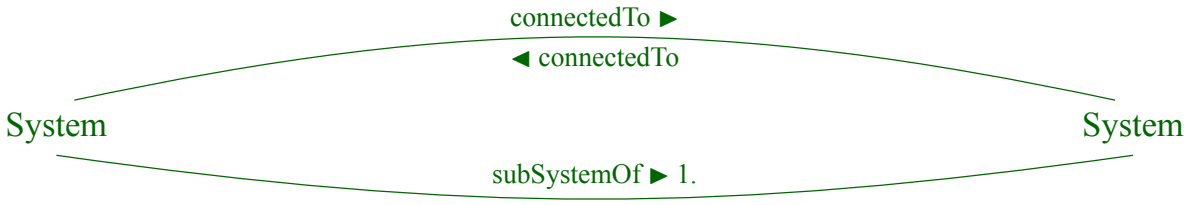

Figure 6: Module seas:SystemOntology: connected systems and sub-systems.

A system may be connected to other systems that are part of its environment. This is modeled by property seas:connectedTo, which is symetric. For example,

<electric_vehicle> seas:connectedTo <electric_vehicle_service_equipment> .

Connected systems interact in some ways. The exact meaning of interact is defined by sub properties of seas:connectedTo. For example, for the electricity to directly flow between an electric vehicle service equipment <electric_vehicle_service_equipment> and an electric vehicle <electric_vehicle>, then they must be linked by property seas:exchangesElectricityWith:

<electric_vehicle> seas:exchangesElectricityWith <electric_vehicle_service_equipment> .

A system can be a sub-system of a unique other system. This is modeled using property seas:subSystemOf, asymetric and functional. For example,

<battery> seas:subSystemOf <electric_vehicle> .

The inverse property of seas:subSystemOf is seas:hasSubSystem.

The properties of a sub-system contribute in some ways to the properties of the super-system. The exact meaning of this contribution must be defined by sub-properties of seas:subSystemOf. For example, if one wants to model that the energy stored in <battery> contribues the the energy stored in <electric_vehicle>, then one may use a sub-property of seas:subSystemOf named seas:subElectricPowerSystemOf:

<battery> seas:subElectricPowerSystemOf <electric_vehicle> .

A connection, modeled by property seas:connectedTo between two systems, describes the potential interactions between connected systems. A connection can be qualified using class seas:Connection. Figure 7 illustrates classes and properties that can be used to qualify connections between systems.

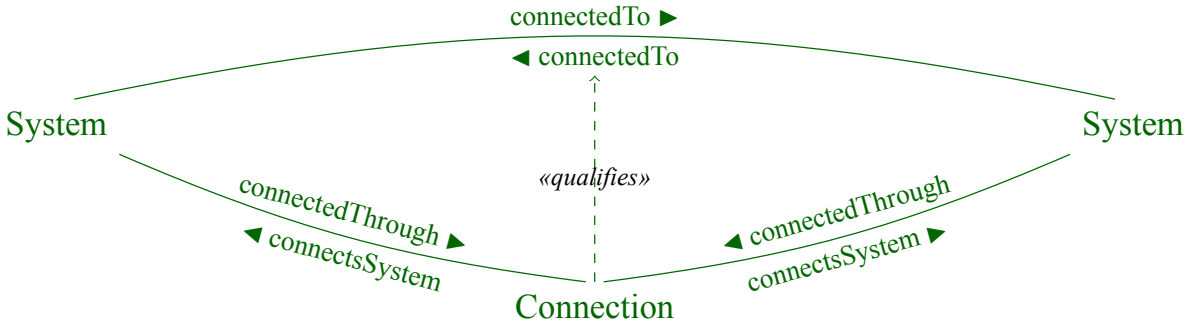

Figure 7: Module seas:SystemOntology: connections between systems.

For example,

<connection> seas:connectsSystem <electric_vehicle> , <electric_vehicle_service_equipment> .
<electric_vehicle> seas:connectedThrough <connection> .
<electric_vehicle_service_equipment> seas:connectedThrough <connection> .

One can then associates this connection with properties that describe the interactions between the connected systems (e.g., exchanged electric power, exchange surface, contact temperature).

A connection point of a system, modeled by properties seas:connectsSystem and seas:connectedThrough, is where a system connects to other systems. A connection point belongs to one and only one system, and can be qualified using the class seas:ConnectionPoint. Figure 8 illustrates the classes and the properties that can be used to qualify connection points of a system.

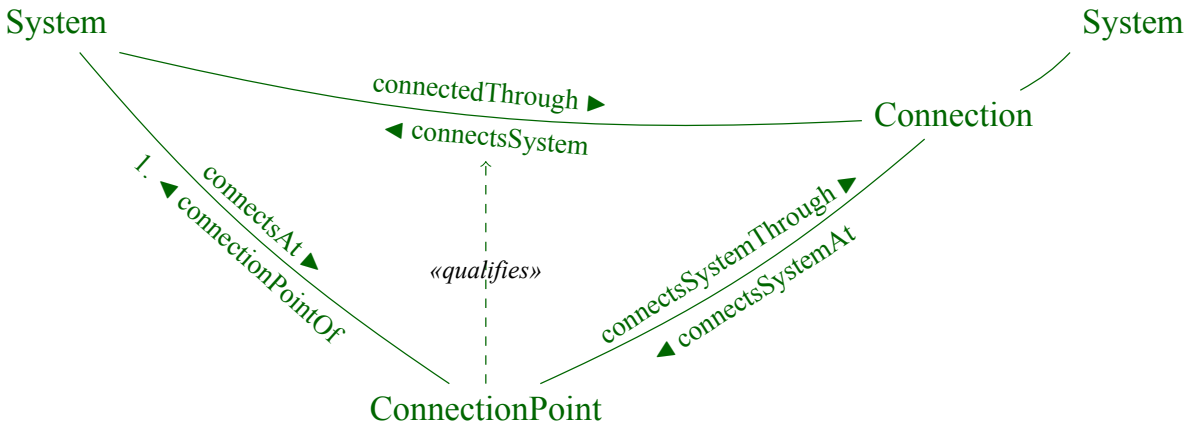

Figure 8: Module seas:SystemOntology: connection points of a system.

For example, a electric vehicle charging station may have three connection points: two plugs of different kind to which electric vehicles can connect, and a three phase connection point to the public grid:

<electric_vehicle> seas:connectsAt <plug_high_voltage> , <normal_plug> ,
<three_phase_connection_point> .

One can then associate a connection point with properties that describe it (e.g., position and speed, voltage and intensity, thermic transmission coefficient).

Instantiating the design pattern This design pattern can be instantiated for different domains. For example to describe zones inside a building (systems), that share a frontier (connections). Properties of systems are typically state variables (e.g., agent population, temperature), whereas properties of connections are typically flows (e.g., heat flow1).

4.2.4 Externalized core module ProcessExecution

IRI: https://w3id.org/pep
Current version IRI: https://w3id.org/pep/pep-1.0

The ProcessExecution ontology is an externalized SEAS module that propose a design pattern that generalizes the core of the W3C Semantic Sensor Network ontology, and the core of the Semantic Actuator Network ontology.

- SSN describes ssn:Sensors that implement ssn:Sensing methods and generate ssn:Observations, which are activities;
- parallel to this, SAN describes san:Actuators that implement san:Acting methods and generate ssn:Actuation activities.

In this ontology, one whishes to generalize these two models, and enable to reuse the same design pattern for other models, notably:

- Web services can execute algorithms.

We propose the following concepts for the ProcessExecution ontology:

pep:ProcessExecutors implement pep:Process methods, and generate pep:ProcessExecutions activities;

As the sensing and acting methods in SSN and SAN, the processes and the process executors can describe required input and outputs. These are modeled using properties pep:hasInput and pep:hasOutput. These properties can also be used to describe the real inputs and outputs for a specific execution.

Alignment with the new version of SSN. The Process Execution ontology is designed to be aligned with the core of the SSN ontology, which is being standardized in the W3C "Spatial Data on the Web" working group. A separate ontology contains this alignment:

IRI: https://w3id.org/pep/SSNAlignment
Current version IRI: https://w3id.org/pep/SSNAlignment-0.9

One can choose between using only the ProcessExecution ontology, or using the ProcessExecution ontology in conjunction with SSN. Furthermore, using this alignment, a RDF graph that is described with SSN become compatible with the ProcessExecution ontology.

Alignment with RDFP. The seas:ProcessExecution ontology is designed to be aligned with the RDFP ontology. A specific ontology contains this alignment:

IRI: https://w3id.org/pep/RDFPAlignment
Current version IRI: https://w3id.org/pep/RDFPAlignment-0.9

One can choose between using only the ProcessExecution ontology, or using the ProcessExecution ontology in conjunction with RDFP. This alignment considers that inputs and outputs are RDF graph descriptions, or directly the URI of a named RDF graph.

4.3 Modules that instantiate the Process design pattern

4.3.1 Module DeviceOntology

IRI: https://w3id.org/seas/DeviceOntology
Current version IRI: https://w3id.org/seas/DeviceOntology-1.0

seas:DeviceOntology imports the Process Execution Platform ontology and the seas:SystemOntology ontologies. This module describes devices, modeled by class seas:Device, as a sub-class of both classes seas:System and pep:ProcessExecutor. As such, they can hence implement pep:Processes, and generate pep:ProcessExecutions. Figure 9 illustrates classes and properties for module DeviceOntology.

Action. Actuators are modeled by class seas:Actuator. This class is equivalent to the class of process executors that implement at least one actuating process, modeled by class seas:Actuating. An actuator can act on multiple properties, and execute multiple actions that act on different properties.

Actuating process executions are modeled by class seas:Actuation. This class is equivalent to the class of process executions that use an actuating method. An actuating process is always executed by an

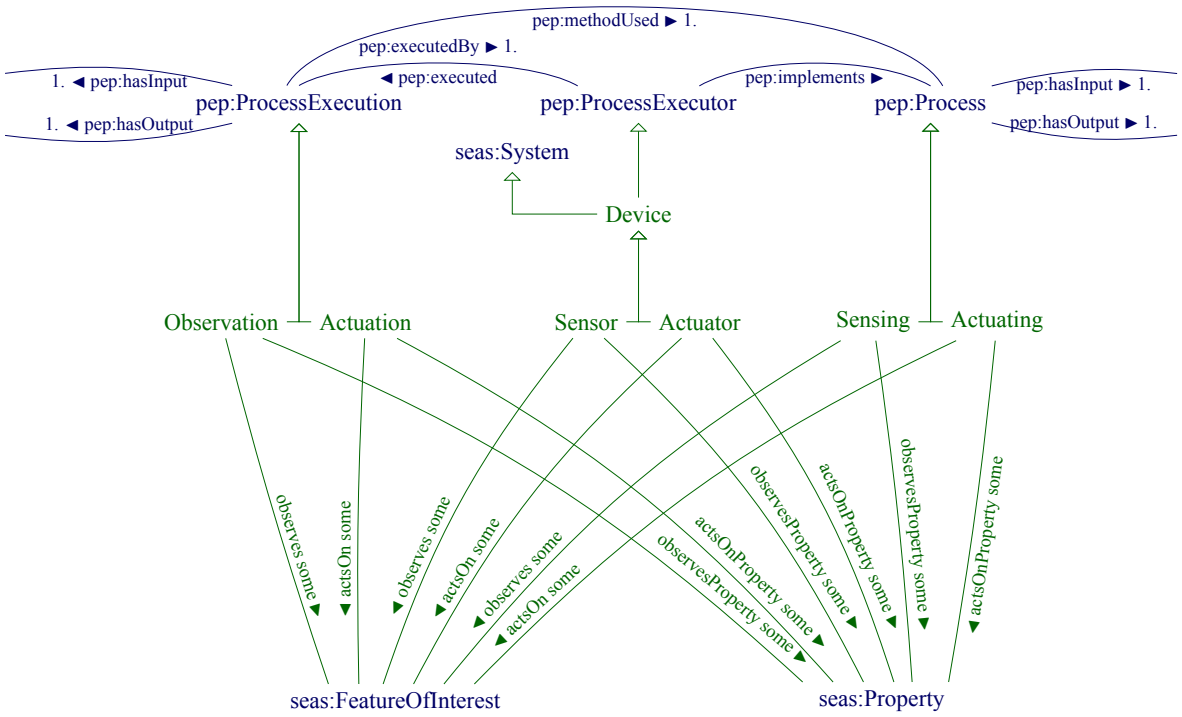

Figure 9: Module seas:DeviceOntology.

actuator, and act on at least one property. An actuation acts on each of the properties the actuating process acts on.

Property seas:actsOnProperty links a seas:Actuating, seas:Actuator, or seas:Actuation to a property on which they act. If an acting acts on a property, then any actuator that implements this acting also acts on this property, and any actuation that used this acting also acts on this property. Furthermore, if an actuator acts on a property, then any actuation executed by this actuator also acts on this property.

Property seas:actsOn links a seas:Actuating, seas:Actuator, or seas:Actuation to a feature of interest on which they act. If x acts on a property, then it also acts on the feature of interest of this property. If an acting acts on a feature of interest, then any actuator that implements this acting also acts on this feature of interest, and any actuation that used this acting also acts on this feature of interest. Furthermore, if an actuator acts on a feature of interest, then any actuation executed by this actuator also acts on this feature of interest.

Observation. Sensors are modeled by class seas:Sensor. This class is equivalent to the class of process executors that implement at least one sensing process, modeled by class seas:Sensing. A sensor can observe multiple properties, and can implement multiple sensing processes. An observation observes each of the properties observed by its sensing process.

Sensing process executions (*observations*) are modeled by class seas:Observation. This class is equivalent to the class of process executions that use a sensing method. A sensing process is always executed by a sensor, and observes at least one property. An observation observes each of the properties the sensing process observes.

Property seas:observesProperty links a seas:Sensing, seas:Sensor, or seas:Observation to a property they observe. If a sensing observes a property, then any sensor that implements this sensing also observes this property, and any observation that used this sensing also observes this property. Furthermore, if a sensor observes a property, then any observation executed by this sensor also observes this property.

Property seas:observes links a seas:Sensing, seas:Sensor, ou seas:Observation to a feature of interest they observe. If x observes a property, then it also observes the feature of interest of this property. If a sensing observes a feature of interest, then any sensor that implements this sensing also observes this feature of interest, and any observation that used this sensing also observes this feature of interest. Furthermore, if a sensor observes a feature of interest, then any observation executed by this sensor also observes this feature of interest.

Design pattern. Sub-classes of seas:Actuator, seas:Actuating, and seas:Actuation may constrain the observed property or its value type, for instance to model the fact the observed property has a boolean value. Module seas:BooleanPropertyOntology illustrates an instance of this design pattern and describes six main classes that constrain the observed/acted on property to be a boolean property. Other ontologies can reuse these definitions to define new sub-classes for their own classification needs.

4.3.2 Module ForecastingOntology

IRI: https://w3id.org/seas/ForecastingOntology
Current version IRI: https://w3id.org/seas/ForecastingOntology-1.0

seas:ForecastingOntology imports the ontology Process Execution Platform ontology and defines:

seas:Forecaster that implement seas:Forecasting processes and do seas:Forecasts.

Sub types of seas:Forecaster, seas:Forecasting, and seas:Forecast, may constrain the type of the value of the property. For instance ontology seas:BooleanOntology may define seas:BooleanForecaster, seas:BooleanForecasting, and seas:BooleanForecast, that constrain the forecasted property to be a seas:BooleanProperty.

Figure 10 illustrates classes and properties of module ForecastingOntology.

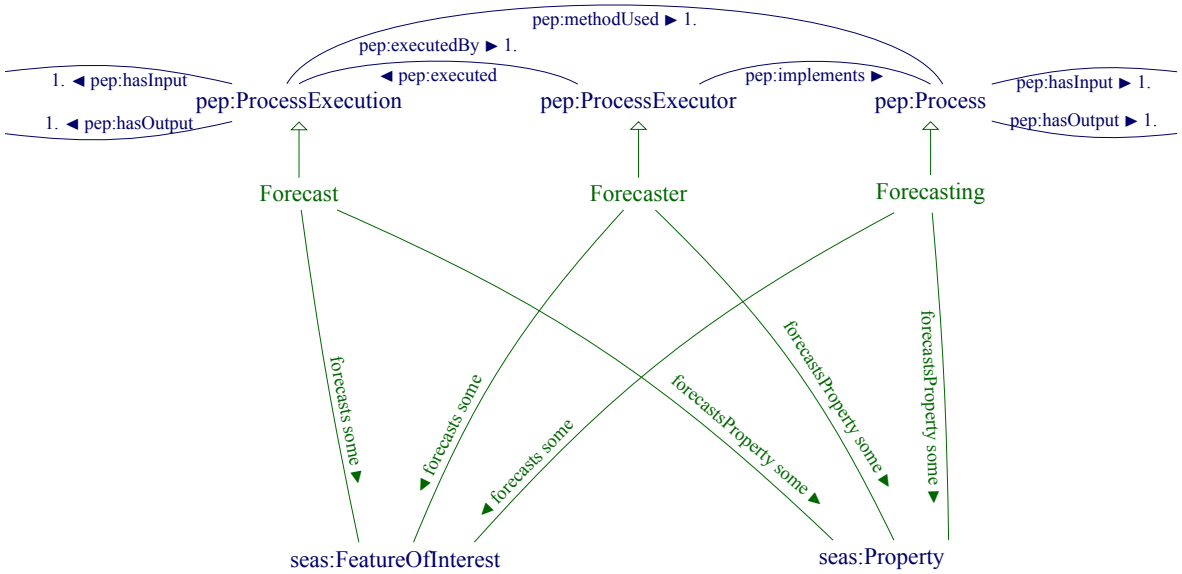

Figure 10: Module seas:ForecastingOntology.

seas:Forecasting. Forecasting is the process of forecasting the state of a property.

seas:Forecaster. A Forecaster implements some Forecasting process, and may generate forecasts.

seas:Forecast. A Forecast is the execution of some Forecasting process by some Forecaster.

seas:forecastsProperty. Links an Forecasting, Forecaster, or Forecast, to the property it forecasts. If a forecasting forecasts a property, then any Forecaster that implements this forecasting also forecasts this property, and any Forecast that used this forecasting also forecasts this property. Furthermore, if a Forecaster forecasts a property, then any Forecast executed by this Forecaster also forecasts this property:

seas:forecastsProperty < pep:implements o seas:forecastsProperty .
seas:forecastsProperty < pep:methodUsed o seas:forecastsProperty .
seas:forecastsProperty < pep:executedBy o seas:forecastsProperty .

seas:forecasts. Links an Forecasting, Forecaster, or Forecast, to the feature of interest a property of which it forecasts. If x forecasts a property, then it also forecasts the feature of interest of this property:

seas:forecastsProperty < seas:forecasts o seas:isPropertyOf .

If a forecasting forecasts a feature of interest, then any Forecaster that implements this forecasting also forecasts this feature of interest, and any Forecast that used this forecasting also forecasts this feature of interest. Furthermore, if a Forecaster forecasts a feature of interest, then any Forecast executed by this Forecaster also forecasts this feature of interest:

seas:forecasts < pep:methodUsed o seas:forecasts .
seas:forecasts < pep:methodUsed o seas:forecasts .
seas:forecasts < pep:executedBy o seas:forecasts .

4.3.3 Module OptimizationOntology

IRI: https://w3id.org/seas/OptimizationOntology
Current version IRI: https://w3id.org/seas/OptimizationOntology-1.0

seas:OptimizationOntology imports the ontology Process Execution Platform ontology and defines:

seas:OptimizationExecutor that implement seas:OptimizationProcess processes and do seas:OptimizationExecutions. Sub types of seas:OptimizationExecutor, seas:OptimizationProcess, and seas:OptimizationExecution, may constrain the type of the value of the property. For instance ontology seas:BooleanOntology may define seas:BooleanOptimizationExecutor, seas:BooleanOptimizationProcess, and seas:BooleanOptimizationExecution, that constrain the optimized property to be a seas:BooleanProperty.

Figure 11 illustrates classes and properties of module OptimizationOntology.

seas:OptimizationProcess. OptimizationProcess is the process of optimizing the state of a property.

seas:OptimizationExecutor. An OptimizationExecutor implements some OptimizationProcess process, and may generate optimizes.

seas:OptimizationExecution. An OptimizationExecution is the execution of some optimization process by some OptimizationExecutor.

seas:optimizesProperty. Links an OptimizationProcess, OptimizationExecutor, or OptimizationExecution, to the property it optimizes. If an OptimizationProcess optimizes a property, then any OptimizationExecutor that implements this OptimizationProcess also optimizes this property, and any OptimizationExecution that used this OptimizationProcess also optimizes this property. Furthermore, if an OptimizationExecutor optimizes a property, then any OptimizationExecution executed by this OptimizationExecutor also optimizes this property:

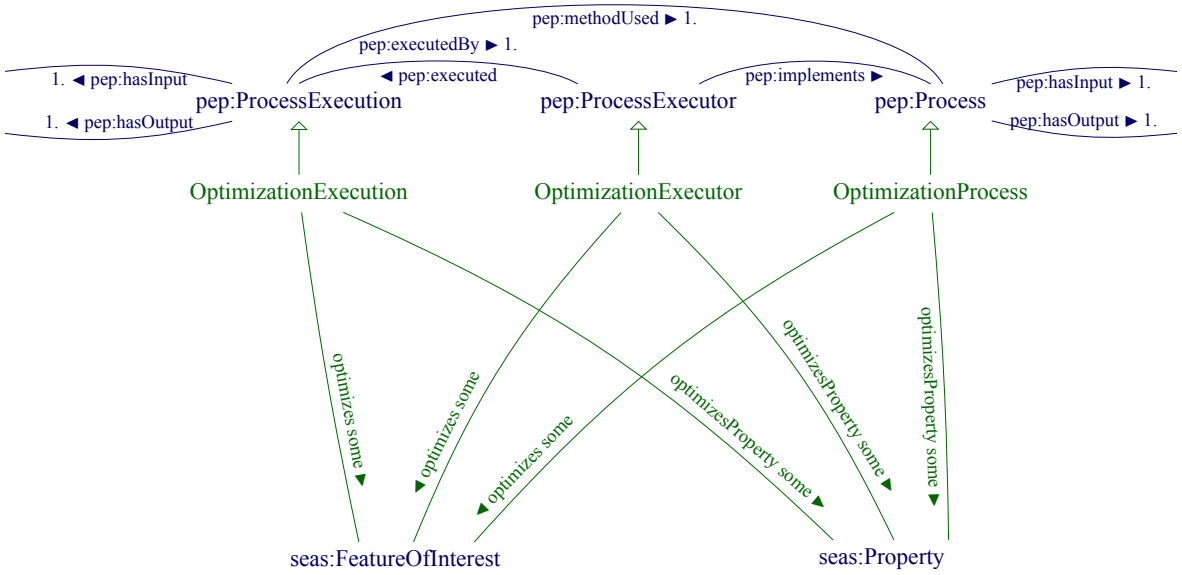

Figure 11: Module seas:OptimizationOntology.

seas:optimizesProperty < pep:implements o seas:optimizesProperty .
seas:optimizesProperty < pep:methodUsed o seas:optimizesProperty .
seas:optimizesProperty < pep:executedBy o seas:optimizesProperty .

seas:optimizes**.** Links an OptimizationProcess, OptimizationExecutor, or OptimizationExecution, to the feature of interest a property of which it optimizes. If x optimizes a property, then it also optimizes the feature of interest of this property:

seas:optimizesProperty < seas:optimizes o seas:isPropertyOf .

If an OptimizationProcess optimizes a feature of interest, then any OptimizationExecutor that implements this OptimizationProcess also optimizes this feature of interest, and any OptimizationExecution that used this OptimizationProcess also optimizes this feature of interest. Furthermore, if a OptimizationExecutor optimizes a feature of interest, then any OptimizationExecution executed by this OptimizationExecutor also optimizes this feature of interest:

seas:optimizes < pep:methodUsed o seas:optimizes .
seas:optimizes < pep:methodUsed o seas:optimizes .
seas:optimizes < pep:executedBy o seas:optimizes .

4.4 Modules that define properties and evaluations

4.4.1 **Module** TimeOntology

IRI: https://w3id.org/seas/TimeOntology
Current version IRI: https://w3id.org/seas/TimeOntology-1.0

seas:TimeOntology imports the ontology seas:EvaluationOntology. This module defines:

- a set of sub-classes of seas:Evaluation that describes the link that exists between the evaluated property, the valued given for the evaluation, and the temporal context of the evaluation.
- means to describe time derivation/sum between properties.
- a set of sub-properties of seas:hasProperty of event and frequency related properties.

Sub-classes of seas:Evaluation

seas:InstantaneousEvaluation**.** The class of evaluations that are relative to an instant.

seas:TimeIntervalEvaluation**.** The class of evaluations that are relative to a proper time interval (i.e., a time interval whose start is different from its end).

seas:TimeVariationEvaluation**.** The given value is the variation of the quantity over the temporal context.

seas:TimeAverageEvaluation**.** The given value is the average of the quantity over the temporal context.

seas:TimeMinimumEvaluation**.** The given value is the minimum of the quantity over the temporal context.

seas:TimeMaximumEvaluation**.** The given value is the maximum of the quantity over the temporal context.

seas:TimeConstantEvaluation**.** The given value is the value of the quantity at any instant of the temporal context.

Time derivations and sums seas:TimeOntology enables to describes the time derivation/sum relationship between two properties.

seas:timeDerivative**.** Links a time relative property to its time derivative property. For example,
<car/1/speed> seas:timeDerivative <car/1/acceleration> .

Means that <car/1/acceleration> is the time derivative of <car/1/speed>.

For the time sum, it is important to qualify the derivation/sum relationship with the time interval over which the sum is computed. Module seas:TimeOntology hence introduces an utility class to describe this:

seas:TimeSum**.** Utility class to qualify the relation between a property and its sum over a time interval. Two sub classes of seas:TimeSum enable to further describe the time interval over which the sum is computed: seas:FixedStartSum, and seas:FixedDurationSum.

seas:summedProperty**.** Links a time sum to the property that is summed over a time interval.

seas:sumProperty**.** Links a time sum to the property that is result of a sum over a time interval.

seas:FixedDurationSum**.** Utility class to qualify the relation between a property and its sum over a time interval whose duration is fixed. The duration over which the sum is computed is specified using property seas:sumDuration. For example:

```
[] a seas:FixedDurationSum ;
   seas:summedProperty <fridge/1/consumptionPower> ;
   seas:sumProperty <fridge/1/consumption> ;
   seas:sumDuration "PT24H"^^xsd:duration .
```

means that at any time, the value of '<fridge/1/consumption>' is the sum of '<fridge/1/consumptionPower>' over the past day.

seas:FixedStartSum**.** "Utility class to qualify the relation between a property and its sum over a time interval whose start is fixed. The start of the time interval over which the sum is computed is specified using property seas:sumStart. For example:

```
[] a seas:FixedStartSum ;
   seas:summedProperty <fridge/1/consumptionPower> ;
   seas:sumProperty <fridge/1/consumption> ;
   seas:sumStart "2014-02-24T00:00:00Z"^^xsd:dateTime .
```

means that at any time, the value of '<fridge/1/consumption>' is the sum of '<fridge/1/consumptionPower>' since February 24th 2014.

Events, frequency seas:TimeOntology introduces a set of sub-properties of seas:hasProperty that enable to classify the link between a feature of interest and a property.

seas:eventNumber**.** Links a feature of interest to a number of events during a temporal context. Functional sub properties of seas:eventNumber define the semantics of the event. For example, if a window was closed 5 times during last night:

```
seas:closingNumber a owl:ObjectProperty , owl:FunctionalProperty ;
  rdfs:subPropertyOf seas:stateChangeNumber .

<window> seas:closingNumber <window/closingNumber> .
<window/closingNumber> seas:evaluation <window/closingNumber/evaluation/1> .
<window/closingNumber/evaluation/1> seas:evaluatedValue 5 ;
  seas:hasTemporalContext [
    time:hasBeginning [ time:asXSDDateTime "2016-09-25T23:00:00Z" ] ;
    time:hasEnd [ time:asXSDDateTime "2016-09-26T07:00:00Z" ] ; ]
```

seas:eventFrequency**.** Links a feature of interest to a frequency of the events during a temporal context. Functional sub properties of seas:eventFrequency define the semantics of the event.

seas:stateChangeFrequency**.** Links a feature of interest to the frequency of experienced property value changes during the considered temporal context. Functional sub properties of seas:stateChangeFrequency define the semantics of the property.

seas:stateDuration**.** Links a feature of interest to the duration during which a property has a given state during the considered temporal context. Functional sub properties of seas:stateDuration define the semantics of the property and the state. For example, if a window was open during 2 hours last night:

```
seas:openDuration a owl:ObjectProperty , owl:FunctionalProperty ;
  rdfs:subPropertyOf seas:stateDuration .

<window> seas:openDuration <window/openDuration> .
<window/openDuration> seas:evaluation <window/openDuration/evaluation/1> .
<window/openDuration/evaluation/1> seas:evaluatedValue "PT2H"^^xsd:duration ;
  seas:hasTemporalContext [
    time:hasBeginning [ time:asXSDDateTime "2016-09-25T23:00:00Z" ] ;
    time:hasEnd [ time:asXSDDateTime "2016-09-26T07:00:00Z" ] ; ]
```

4.4.2 **Module** FlexibilityOntology

IRI: https://w3id.org/seas/FlexibilityOntology
Current version IRI: https://w3id.org/seas/FlexibilityOntology-1.0

seas:FlexibilityOntology imports ontologies seas:EvaluationOntology and seas:PlayerOntology. This module defines properties that are controllable by some business actor, and a set of sub-classes of seas:Evaluation to describe the flexibility of these properties.

Property

seas:ControllableProperty. A property that is controllable and can be operated on.

seas:controlledBy. Links a property to a seas:Player that can control it.

Sub-classes of seas:Evaluations

seas:CapabilityEvaluation. The amount of the property that the feature of interest can produce or deal with. The quantity dimension is that of the controllable property.

seas:ReductionFlexibilityEvaluation. An evaluation of the reduction flexibility of a controllable property.

seas:AugmentationFlexibilityEvaluation. An evaluation of the augmentation flexibility of a controllable property.

seas:CutFlexibilityEvaluation. An evaluation of the augmentation flexibility of a controllable property.

seas:MoveFlexibilityEvaluation. An evaluation of the move flexibility of a controllable property.

seas:ReductionFlexibilityEvaluation. An evaluation of the reduction flexibility of a controllable property.

4.4.3 Module OperatingOntology

IRI: https://w3id.org/seas/OperatingOntology
Current version IRI: https://w3id.org/seas/OperatingOntology-1.0

seas:OperatingOntology imports the ontology seas:EvaluationOntology, and defines evaluations of operating features of interest.

Sub-classes of seas:Evaluations

seas:MinimumOperatingEvaluation. The evaluated value is the minimum operating value of the property.

seas:NominalOperatingEvaluation. The evaluated value is the nominal operating value of the property.

seas:MaximumContinuousOperatingEvaluation. The evaluated value is the maximum continuous operating value of the property.

seas:MaximumPeakRatingEvaluation. The evaluated value is the maximum peak rating value of the property.

4.4.4 Module FailableSystemOntology

IRI: https://w3id.org/seas/FailableSystemOntology
Current version IRI: https://w3id.org/seas/FailableSystemOntology-1.0

seas:FailableSystemOntology imports the ontology seas:SystemOntology, and describes systems that can fail, and their properties.

Property

seas:failure. Links a feature of interest to its failure property, which may further be quantified in terms of time, duration, cost, ...

seas:repair. Links a feature of interest to its repair property, which may further be quantified in terms of time, duration, cost, ...

seas:replace. Links a feature of interest to its replace property, which may further be quantified in terms of time, duration, cost, ...

seas:checkup. Links a feature of interest to its checkup property, which may further be quantified in terms of time, duration, cost, ...

seas:unavailability. Links a feature of interest to its unavailability property, which may further be quantified in terms of time, duration, cost, ...

4.4.5 Module ComfortOntology

IRI: https://w3id.org/seas/ComfortOntology
Current version IRI: https://w3id.org/seas/ComfortOntology-1.0

seas:ComfortOntology imports the ontology seas:EvaluationOntology, and describes evaluations related to the comfort of some agent (foaf:Agent). Given a property and an agent, one may specify that some evaluation of the property is relative to an agent, and that it is comfortable, uncomfortable, or critically uncomfortable for the agent.

Sub-classes of seas:Evaluations

seas:AgentComfortEvaluation. An agent comfort evaluation is a seas:Evaluation of a seas:Property that is relative to a specific agent. The property impacts the comfort of the agent. Formally, it is equivalent to the set of evaluations that are linked by property seas:relativeToAgent to some foaf:Agent.

seas:relativeToAgent. Links an evaluation to the agent (foaf:Agent) whose comfort the property impacts.

seas:ComfortableEvaluation. A comfortable evaluation is an evaluation of the value for a property that is considered comfortable for the associated agent.

- seas:MinimumComfortableEvaluation. A minimum comfortable evaluation is an evaluation of the minimum value for a property that is considered comfortable for the associated agent.
- seas:MaximumComfortableEvaluation. A maximum comfortable evaluation is an evaluation of the maximum value for a property that is considered comfortable for the associated agent.

seas:UncomfortableEvaluation. An uncomfortable evaluation is an evaluation of the value for a property that is considered uncomfortable for the associated agent.

- seas:MinimumUncomfortableEvaluation. A minimum uncomfortable evaluation is an evaluation of the minimum value for a property that is considered uncomfortable for the associated agent.
- seas:MaximumUncomfortableEvaluation. A maximum uncomfortable evaluation is an evaluation of the maximum value for a property that is considered uncomfortable for the associated agent.

seas:CriticallyUncomfortableEvaluation**.** A critically uncomfortable evaluation is an evaluation of the value for a property that is considered critically uncomfortable for the associated agent. It may be dangerous for the agent.

seas:MinimumCriticallyUncomfortableEvaluation**.** A minimum critically uncomfortable evaluation is an evaluation of the minimum value for a property that is considered critically uncomfortable for the associated agent. It may be dangerous for the agent.

seas:MaximumCriticallyUncomfortableEvaluation**.** A maximum critically uncomfortable evaluation is an evaluation of the maximum value for a property that is considered critically uncomfortable for the associated agent. It may be dangerous for the agent.

4.4.6 **Module** GreenKPIOntology

IRI: https://w3id.org/seas/GreenKPIOntology
Current version IRI: https://w3id.org/seas/GreenKPIOntology-1.0

seas:GreenKPIOntology imports the ontology seas:FeatureOfInterestOntology. This module defines properties that describe the "Green" aspect of a feature of interest.

Functional sub-properties of seas:hasProperty

seas:local**.** Percentage of the feature of interest that is local.

seas:carbonFootprint**.** The carbon footprint of the feature of interest.

4.4.7 **Module** PeriodicSignalOntology

IRI: https://w3id.org/seas/PeriodicSignalOntology
Current version IRI: https://w3id.org/seas/PeriodicSignalOntology-1.0

seas:PeriodicSignalOntology describes periodic signals properties.

Properties

seas:PeriodicProperty The class of properties that evolve periodically in time.

Functional sub-properties of seas:hasProperty

seas:frequency Links a periodic property to its frequency.

Sub-classes of seas:Evaluation

seas:PeakEvaluation The evaluated value is the Peak of the periodic signal.

seas:THDEvaluation The evaluated value is the total harmonic distortion of the periodic signal.

seas:RMSAmplitudeEvaluation The evaluated value is the Root Mean Square amplitude of the periodic signal.

4.4.8 **Module** ComplexOntology

IRI: https://w3id.org/seas/ComplexOntology
Current version IRI: https://w3id.org/seas/ComplexOntology-1.0

seas:ComplexOntology imports the ontology seas:FeatureOfInterestOntology. This ontology defines properties whose values may be represented as mathematical complexes. It also defines common properties of such properties such as the real or the imaginary part.

seas:ComplexProperty. A property that is quantifiable by a complex value with a quantity dimension.

Functional sub-properties of seas:hasProperty

seas:real The real part of a complex property.

seas:imaginary The imaginary part of a complex property.

seas:module The module of a complex property.

seas:phase The phase of a complex property.

4.4.9 **Module** StatisticsOntology

IRI: https://w3id.org/seas/StatisticsOntology
Current version IRI: https://w3id.org/seas/StatisticsOntology-1.0

seas:StatisticsOntology describes the statistical distribution of values of some property.

Sub-classes of seas:Evaluation

seas:DistributionMinimumEvaluation The evaluated value is the minimum of the property value distribution.

seas:DistributionMaximumEvaluation The evaluated value is the maximum of the property value distribution.

seas:DistributionMeanEvaluation The evaluated value is the mean of the property value distribution.

seas:DistributionMedianEvaluation The evaluated value is the median of the property value distribution.

seas:StandardDeviationEvaluation The evaluated value is the standard deviation of the property value distribution.

seas:RankedDistributionEvaluation The evaluated value is to be interpreted with respect to a rank, which is given as the object of property seas:rank.

- seas:rank Rank number of a ranked distribution evaluation, such as for quartiles, deciles or percentiles.
- seas:QuartileEvaluation The evaluated value is one of the quartiles of the property value distribution. The rank of the quartile is the object of property seas:rank.
- seas:DecileEvaluation The evaluated value is one of the deciles of the property value distribution. The rank of the decile is the object of property seas:rank.
- seas:PercentileEvaluation The evaluated value is one of the percentiles of the property value distribution. The rank of the percentile is the object of property seas:rank.

4.5 Vertical modules for the Smart Grid and Micro Grid domains

4.5.1 **Module** PlayerOntology

IRI: https://w3id.org/seas/PlayerOntology
Current version IRI: https://w3id.org/seas/PlayerOntology-1.0

seas:PlayerOntology imports the ontology Process Execution Platform ontology and the ontology seas:SystemOntology. This module defines business players for the SEAS project. A business player can offer services, and be paid to execute services.

Systems

seas:Player. One of the important people, companies etc involved in a particular industry, market, situation etc. (source: the Longman Business Dictionary). Is both a seas:System, and a pep:ProcessExecutor.

seas:Operator. An Operator is an actor that regulates operations of players in the system.

seas:Aggregator. The class of Aggregator SEAS players. An entity responsible for planning, scheduling, accounting, billing, and settlement for energy deliveries from the aggregator's portfolio of seller and/or buyers. Aggregators seek to bring together customers or generators so they can buy or sell power in bulk, making a profit on the transaction.

seas:Authority. The class of Authority SEAS players.

seas:ElectricityTrader. The class of players that trade electricity.

seas:SmartChargingProvider. The class of SmartChargingProvider SEAS players.

seas:BalanceResponsibleParty. The class of BalanceResponsibleParty SEAS players. A party that has a contract proving financial security and identifying balance responsibility with the Imbalance Settlement Responsible of the Market Balance Zone entitling the party to operate in the market. This is the only role allowing a party to nominate energy on a wholesale level.

seas:BalanceServiceProvider. The class of BalanceServiceProvider SEAS players. Also called Balance Responsible Parties (BRP). The balance responsible party is the only role allowing a party to buy or sell energy on a wholesale level. The balance responsible party must declare to the TSO the production and consumption plans for the next day on its balance perimeter. It undertakes to pay financial compensation to the TSO for negative imbalances (production-consumption) and is financially compensated for positive imbalances.

seas:ChargeServiceProvider. The class of ChargeServiceProvider SEAS players. The CSP offers e-mobility services to Electric Vehicle users (may include charging, search & find, routing and other services). It operates a contract party for the EV user, taking care of the authentication and billing process. It provides an access card available for many EVSE whose CSO have an agreement with the CSP, and may have some roaming agreement with other CSP registered by a clearing house.

seas:ChargingStationOperator. The class of ChargingStationOperator SEAS players. A Charging Station Operator is a party delivering and managing physical equipment to supply the charging process of the electrical vehicle (EVSE). The CSO is generally investor, owner and operator of EVSE and the private electricity network to which they are connected : which is defined as the charging station.

seas:ClearingHouse. The class of ClearingHouse SEAS players. The clearing house records all the roaming agreements between CSP and CSO (EV service roaming). It facilitates data exchange between roaming partners : authentication, validation of contracts, charge retail records (duration, energy, load...)

seas:CurtailmentServiceProvider. The class of CurtailmentServiceProvider SEAS players. A company that serves as an intermediary between utilities and customers, pooling together groups of customers who participate in demand response programs to reduce energy usage during periods of peak demand. Aggregate load profiles of small and medium consumers to have a better support for the participation in DR events.

seas:DataBroker. The class of DataBroker SEAS players. An entity that collects data from a variety of sources, including Internet and online sources as well as databases, print documentation, and surveys to package and sell as a product or service to other entities. It can include personal consumer data or business data to serve information needs of private sector and governmental agencies, and could include information generated by Smart Grid or M2M applications. Also known as an information broker.

seas:DataManagementSystem. The class of DataManagementSystem SEAS players. Retrieves, processes and recovers data.

seas:DistributedEnergyResourcesInformationProvider. The class of DistributedEnergyResourcesInformationProvider SEAS players. Provides information of power system variables such as loads and production from renewables, forecasts, information on electric vehicles etc... It can act as a trusted third party responsible for dispatching the consumption between many energy suppliers and sharing a registry for metering data (electric roaming).

seas:DistributionSystemOperator. The class of DistributionSystemOperator SEAS players. Has the responsibility of operating the distribution network, ensuring the maintenance and development of the infrastructure and the respect of the quality of supply (reduced disconnection time, voltage within bands, etc...). The DSO is also responsible for the metering.

seas:TransmissionSystemOperator. A transmission system operator (TSO) is an entity entrusted with transporting energy in the form of natural gas or electrical power on a national or regional level, using fixed infrastructure. (source: Wikipedia).

seas:Consumer. The class of Consumer SEAS players. Person for whom something was ultimately created or intended. NOTE: legal issues in dealing with privacy and market legislation.

seas:EnergyEndCustomer. The class of EnergyEndCustomer SEAS players. Consumer, prosumer, End-user, Home-User, EV user, Buyer, Seller.

seas:EnergyProducerOperator. The class of EnergyProducerOperator SEAS players. Produces energy. Either buys, sells or “lends” energy. Manages the electricity production and sells the electricity to the electricity market. He gives information (forecast (production, weather), real time production, price...).

seas:EnergyProvider. The class of Energy Provider SEAS players.

seas:EnergyRetailer. The class of EnergyRetailer SEAS players. Sell or Buy Energy to the End-Customer and purchases it on the electricity market. It charges the customer based on the flexibility, duration and power. It communicates to the customer the energy metering values, in accordance with the DSO or TSO metering.

seas:ForecastProvider. The class of ForecastProvider SEAS players. Provides through forecasts the value of power system variables such as loads and production from renewable energy or performs and updates the forecast values, which are transmitted to the network operator. Kind of value : weather, price, consumption, generation.

seas:GenerationEquipment. The class of GenerationEquipment SEAS players. A customer of the network injecting electricity. Examples found in the different UCs are PV producer, CHP, and Wind producer.

seas:HomeAndBuildingManagementSystem. The class of HomeAndBuildingManagementSystem SEAS players. A Home and Building Manager System is a hardware device with communication ability, allowing access to a residential home or building and offering energy related services based on actions/command and data collection/restitution. Connects in-home digital devices, such as PCs, mobile phones, entertainment technology, thermostats, home security systems and smart appliances, into a common network. Extracts useful knowledge about consumption behavior to be used in management methodologies, balancing demand and supply and possibly considering renewable sources available at buildings.

Connections

seas:ElectricityMarket. The class of Electricity Market SEAS players

seas:ElectricityCapacityMarket. Electricity Capacity markets provide an additional incentive for developers and owners of generating capacity (i.e. power plants or demand response providers) to make their capacity available to electric markets where price signals alone would not. Capacity providers are paid on a kilowatt per year basis for the capacity that a power plant can generate or, in the case of demand response, the capacity of power that can be reduced.

seas:WholesaleElectricityMarket. Wholesaling is the sale of goods or merchandise to retailers; to industrial, commercial, institutional, or other professional business users; or to other wholesalers and related subordinated services. In general, it is the sale of goods to anyone other than a standard consumer.

seas:LongTermElectricityMarket. Market where electricity is traded long before it is delivered.

seas:IntradayElectricityMarket. The Intraday market offers the opportunity to continuously trade power products in hourly intervals as well as freely definably block orders up to 5 minutes prior to delivery.

seas:DayAheadElectricityMarket. The day-ahead market is an auction where power is traded for delivery during the next day.

4.5.2 Module EnergyFormOntology

IRI: https://w3id.org/seas/EnergyFormOntology
Current version IRI: https://w3id.org/seas/EnergyFormOntology-1.0

seas:EnergyFormOntology imports seas:FeatureOfInterestOntology, and defines instances of energy forms.

Classes

seas:EnergyForm. The class of energy forms.

seas:RenewableEnergyForm. The class of renewable energy forms.

seas:UnrenewableEnergyForm. The class of unrenewable energy forms.

Instances

seas:KineticEnergy The energy of a body or a system with respect to the motion of the body or of the particles in the system.

seas:PotentialEnergy The energy of a body or a system with respect to the position of the body or the arrangement of the particles of the system.

seas:MechanicalEnergy The power that an object gets from its position and motion.

seas:MechanicalWaveEnergy The power that an object gets from energy propagated by a material's oscillations.

seas:ChemicalEnergy The energy that is contained in molecules.

seas:ElectricEnergy The energy that is from electric fields.

seas:MagneticEnergy The energy that is from magnetic fields.

seas:RadiantEnergy The energy transmitted in wave motion, especially electromagnetic wave motion.

seas:NuclearEnergy The energy released by reactions within atomic nuclei, as in nuclear fission or fusion.

seas:IonizationEnergy The amount of energy required to remove an electron from an atom to form a cation; also called ionization potential.

seas:ElasticEnergy potential energy that is stored when a body is deformed (as in a coiled spring).

seas:GravitationalEnergy The gravitational potential energy is energy an object possesses because of its position in a gravitational field.

seas:RestEnergy The energy equivalent to the mass of a particle at rest in an inertial frame of reference, equal to the rest mass times the square of the speed of light.

seas:ThermalEnergy Thermal energy is the energy that is generated and measured by heat.

seas:HeatEnergy The transfer of energy from one thing to another by kinetic energy, usually causing a higher temperature.

4.5.3 **Module** ElectricPowerSystemOntology

IRI: https://w3id.org/seas/ElectricPowerSystemOntology
Current version IRI: https://w3id.org/seas/ElectricPowerSystemOntology-1.0

seas:ElectricPowerSystemOntology imports ontologies seas:SystemOntology and seas:EvaluationOntology, and defines:

1. electric power systems that consume, produce, or store electricity,
2. electrical connections between electric power systems, where electricity is exchanged,
3. electrical connection Points of electric power systems, through which electricity flows in/out the power systems.

This module defines properties related to the electricity: electric power, electric energy, voltage, current, resistance, reactance, inductance, suscpetance.

Systems, connections, connection points

seas:ElectricPowerSystem**.** The class of electric power systems, i.e., systems that exchange electricity (i.e., electric power) with other systems. Electric power systems may consume, produce, or store electricity. Such systems are defined by subclasses of electric power system. Electric power systems that dissipate energy are thermodynamic systems (i.e., they can then dissipate heat). Electric power systems can exchange electricity with other electric power systems. These exchange are only possible through electrical connections between electric power systems. Electric power systems can have sub power systems:

- The consumed or produced electricity of subsystems contribute to the consumed or produced electricity of the supersystem;
- the stored electricity of subsystems contribute to the stored electricity of super systems.

seas:exchangesElectricityWith**.** Links an electric power system to another electric power system it exchanges electricity with. This property can be qualified using class seas:ElectricalConnection, which connects (at least) the two electric power systems. For example, for the electricity to flow between a battery and a light, then they must share an electrical connection:

```
<battery> a seas:ElectricPowerSystem .
<light> a seas:ElectricPowerSystem .
<battery> seas:exchangesElectricityWith <light> .
```

If there is an electrical connection between several electric power systems, then one may infer these electric power systems can exchange electricity.

seas:subElectricPowerSystemOf**.** Links an electric power system to its super electric power systems. The electricity of a sub power system contributes to the electricity of its super electric power system. For example,

- The consumed electricity of an electric power comsumer system contributes to the consumed electricity of its super electric power consumer system;
- The produced electricity of an electric power producer system contributes to the produced electricity of its super electric power producer system;
- the stored electricity of an electric power storage system contributes to the stored electricity of its super electric power storage system.

seas:ElectricalConnection**.** The class of electrical connections between electric power systems. An electrical connection describes potential electricity flows between the electric power systems it connects. For example, the following RDF graph states that a power bus connects two power lines and a transformer:

```
<power_bus> a seas:ElectricalConnection ;
  seas:connectsSystem <power_line_1> , <transformer> , <power_line_2> .
```

Only electric power systems and electrical connection points are connected through an electrical connection. A connection between electric power systems is not necessarily an electrical connection. Any electric power system that exchanges electricity through an electrical connection is connected at least through one of its power connection points to the electrical connection:

If:

<power_system> a seas:ElectricPowerSystem .
<power_connection> a seas:ElectricalConnection .
<power_system> seas:connectedThrough <power_connection> .

then there exists _:powerConnectionPoint such that:

_:powerConnectionPoint a seas:ElectricalConnectionPoint ;
seas:connectionPointOf <power-system> ;
seas:connectsSystemThrough <power_connection> .

seas:ElectricalConnectionPoint. The class of electrical connection points of electric power systems, at which they may be electrically connected to other systems. For example, an electric vehicle service equipment may have three electric power connection points: two different kinds of plugs that enable to charge electric vehicles, and a three-phase power bus connection point to the grid:

<electric_vehicle> a seas:ElectricPowerSystem ;
seas:connectsAt <plug_high_voltage> , <normal_plug> , <three_phase_connection_point> .
<plug_high_voltage> a seas:ElectricalConnectionPoint .
<normal_plug> a seas:ElectricalConnectionPoint .
<three_phase_connection_point> a seas:ElectricalConnectionPoint .

Any system connected through an electrical connection is connected at one of its electrical connection points to the electrical connection: if:

<system> a seas:System .
<system> seas:connectedThrough <connection> .
<connection> a seas:ElectricalConnection .

then there exists _:connectionPoint such that:

_:connectionPoint a seas:ElectricalConnection Point ;
seas:connectionPointOf <system> ;
seas:connectsSystemAt <connection> .

An electrical connection point describes an electric signal, and the electricity that enters/leaves the electric power system. An electrical connection point only belongs to an electric power system, and connects it through electrical connections. A connection point of an electric power system is not necessarily an electrical connection.

Sub-classes of seas:ElectricPowerSystem

seas:ElectricPowerConsumer An electric power consumer is an electric power system that is capable of consuming electricity.

seas:ElectricPowerProducer An electric power consumer is an electric power system that is capable of producing electricity.

seas:ElectricPowerStorageSystem An electric power storage system is an electric power system that is capable of storing electricity. Usually a battery, an electric vehicle, or a microgrid.

seas:ElectricPowerTransmissionSystem An electric power transmission system is an electric power system that is capable of transmitting electricity. Usually to transmit electricity on a certain distance within an electric power network, or to transform it.

seas:ElectricPowerLine An electric power line is an electric transmission system that is capable of transmitting electricity on a certain distance within an electric power network. An electric power line must be connected to two power connections, although this requirement cannot be modeled using OWL axioms if we want to remain in a decidable fragment.

seas:ElectricPowerTransformer An electric power transformer is an electric power system that is capable of transforming electricity within a power network, between a primary connection point and a secondary connection point.

seas:primarilyConnectedThrough Links an electric power transformer to its primary connection

seas:secondarilyConnectedThrough Links an electric power transformer to its primary connection

seas:connectsPrimarilyAt Links an electric power transformer to its primary connection

seas:connectsSecondarilyAt Links an electric power transformer to its primary connection

Sub-classes of seas:ElectricalConnection

seas:DirectCurrentConnection The class of electrical connections through which energy flows using direct current.

seas:TwoWireDirectCurrentPowerBus A two-wire direct current power bus is a connection composed of two-wires (potentially plus the protective earth): wire for the positive; wire for the negative.

seas:ThreeWireDirectCurrentPowerBus A three-wire direct current power bus is a connection composed of three wires (potentially plus the protective earth): wire for the positive; wire for the mid-wire; wire for the negative.

seas:AlternatingCurrentConnection The class of electric power connections through which energy flows using alternating current.

seas:ThreePhasePowerBus A three-phase power bus is a connection composed of four wires (plus the protective earth): wires R, S, T, for the phases; wire N for the neutral.

seas:connectsSystemInStar Links a three-phase power bus to one of the electric power systems it connects with a star configuration.

seas:connectedInStarThrough Links an electric power system to a three-phase power bus connection with which it is connected with a star configuration.

seas:connectsSystemInTriangle Links an electric power system to a three-phase power bus connection with which it is connected with a star configuration.

seas:connectedInTriangleThrough Links an electric power system to a three-phase power bus connection with which it is connected with a triangle configuration.

seas:SinglePhasePowerBus A single phase power bus is a connection composed of two-wires (plus the protective earth): wire for the phase; wire for the neutral.

seas:connectsSystemInRN Links a three-phase power bus to one of the electric power systems it connects with a RN configuration.

seas:connectedInRNThrough Links an electric power system to a three-phase power bus connection with which it is connected with a RN configuration.

seas:connectsSystemInSN Links an electric power system to a three-phase power bus connection with which it is connected with a SN configuration.

seas:connectedInSNThrough Links an electric power system to a three-phase power bus connection with which it is connected with a SN configuration.

seas:connectsSystemInTN Links a three-phase power bus to one of the electric power systems it connects with a TN configuration.

seas:connectedInTNThrough Links an electric power system to a three-phase power bus connection with which it is connected with a TN configuration.

seas:SplitPhasePowerBus A split-phase power bus is a connection composed of three wires (plus the protective earth): wire for the hot 1 phase; wire for the neutral; wire for the hot 2 phase.

- seas:connectsSystemInHot1N Links a split-phase power bus to one of the electric power systems it connects with a hot1-to-N configuration.
- seas:connectedInHot1NThrough Links an electric power system to a split-phase power bus connection with which it is connected with a hot1-to-N configuration.
- seas:connectsSystemInHot2N Links a split-phase power bus to one of the electric power systems it connects with a hot2-to-N configuration.
- seas:connectedInHot2NThrough Links an electric power system to a split-phase power bus connection with which it is connected with a hot2-to-N configuration.
- seas:connectsSystemInHot1Hot2 Links a split-phase power bus to one of the electric power systems it connects with a hot1-to-hot2 configuration.
- seas:connectedInHot1Hot2Through Links an electric power system to a split-phase power bus connection with which it is connected with a hot1-to-hot2 configuration.
- seas:connectsSystemInHot2Hot1 Links a split-phase power bus to one of the electric power systems it connects with a hot2-to-hot1 configuration.
- seas:connectedInHot2Hot1Through Links an electric power system to a split-phase power bus connection with which it is connected with a hot2-to-hot1 configuration.

Sub-classes of seas:ElectricalConnectionPoint

seas:DirectCurrentConnectionPoint The class of electrical connection points through which energy flows using direct current.

- seas:TwoWireDirectCurrentConnectionPoint A two-wire direct current connection point is a connection point composed of two-wires (potentially plus the protective earth): wire for the positive; wire for the negative.
- seas:ThreeWireDirectCurrentConnectionPoint A three-wire direct current power connection point is a connection point composed of three wires (potentially plus the protective earth): wire for the positive; wire for the mid-wire; wire for the negative.

seas:AlternatingCurrentConnectionPoint The class of electric power connection points through which energy flows using alternating current.

- seas:ThreePhaseConnectionPoint A three-phase connection point is a connection point composed of four wires (plus the protective earth): wires R, S, T, for the phases; wire N for the neutral.
  - seas:connectsSystemInStarAt Links a three-phase power bus to one of the three-phase connection points it connects with a star configuration.
  - seas:connectsSystemInStarThrough Links a three-phase connection point to a three-phase power bus connection with which it is connected with a star configuration.
  - seas:connectsSystemInTriangleAt Links a three-phase power bus to one of the three-phase connection points it connects with a triangle configuration.
  - seas:connectsSystemInTriangleThrough Links a three-phase connection point to a three-phase power bus connection with which it is connected with a triangle configuration.
- seas:SinglePhaseConnectionPoint A single phase connection point is a connection point composed of two-wires (plus the protective earth): wire for the phase; wire for the neutral.
  - seas:connectsSystemInRNAt Links a three-phase power bus to one of the single phase connection points it connects with a SN configuration.

seas:connectsSystemInRNThrough Links a single phase connection point to a three-phase power bus connection with which it is connected with a RN configuration.

seas:connectsSystemInSNAt Links a three-phase power bus to one of the single phase connection points it connects with a SN configuration.

seas:connectsSystemInSNThrough Links a single phase connection point to a three-phase power bus connection with which it is connected with a SN configuration.

seas:connectsSystemInTNAt Links a three-phase power bus to one of the single phase connection points it connects with a TN configuration.

seas:connectsSystemInTNThrough Links a single phase connection point to a three-phase power bus connection with which it is connected with a TN configuration.

Functional sub-properties of seas:hasProperty

For transformers.

seas:tapChanger The Tap changer to change the voltage regulation of the transformer.

Link to property seas:CurrentProperty**.** The class of properties that are quantifiable and use a quantity dimension of electric current.

seas:electricCurrent Links an electrical connection point to one of its current properties. There may be many current properties of a connection point, so this property is not functional.

seas:positiveWireCurrent Links a direct current connection point to the current in its positive wire. By convention, the value is positive if it enters the system.

seas:midWireCurrent Links a direct current connection point to the current in its mid-wire. By convention, the value is positive if it enters the system.

seas:negativeWireCurrent Links a direct current connection point to the current in its negative wire. By convention, the value is positive if it enters the system.

seas:phaseWireCurrent Links a single phase connection point to the current in its phase wire. By convention, the value is positive if it enters the system.

seas:neutralWireCurrent Links a single phase connection point to the current in its neutral wire. By convention, the value is positive if it leaves the system.

seas:rCurrent Links a three-phase connection point to the current in its R wire. By convention, the value is positive if it enters the system.

seas:sCurrent Links a three-phase connection point to the current in its S wire. By convention, the value is positive if it enters the system.

seas:tCurrent Links a three-phase connection point to the current in its T wire. By convention, the value is positive if it enters the system.

seas:nCurrent Links a three-phase connection point to the current in its N wire. By convention, the value is positive if it enters the system.

Link to property seas:VoltageProperty**.** The class of properties that are quantifiable and use a quantity dimension of electric potential.

seas:voltage Voltage is the voltage at some electrical connection, or at some electrical connection point. There may be many voltage properties of an electrical connection or an electrical connection point, so this property is not functional.

seas:positiveToMidVoltage Links a direct current connection or a direct current connection point to the voltage between its positive and its mid wire.

seas:positiveToNegativeVoltage Links a direct current connection or a direct current connection point to the voltage between its positive and its negative wire.

seas:midToNegativeVoltage Links a direct current connection or a direct current connection point to the voltage between its mid and its negative wire.

seas:phaseToNeutralVoltage Links a single phase power bus or a single phase connection point to the voltage between its phase and neutral wires.

seas:RSVoltage Links a three-phase power bus or a three-phase connection point to the voltage between its R and its S wires.

seas:STVoltage Links a three-phase power bus or a three-phase connection point to the voltage between its S and its T wires.

seas:TRVoltage Links a three-phase power bus or a three-phase connection point to the voltage between its T and its R wires.

seas:RNVoltage Links a three-phase power bus or a three-phase connection point to the voltage between its R and its N wires.

seas:SNVoltage Links a three-phase power bus or a three-phase connection point to the voltage between its S and its N wires.

seas:TNVoltage Links a three-phase power bus or a three-phase connection point to the voltage between its T and its N wires.

Link to property seas:ElectricPowerProperty**.** The class of properties that are quantifiable and have a quantity kind of electric power.

seas:electricPower Links a feature of interest to one of its electric power properties. There may be many power properties for a feature of interest, so this property is not functional.

seas:consumedElectricPower Links an electric power system to the consumed electric power.

seas:producedElectricPower Links an electric power system to the produced electric power.

seas:storageElectricPower Links an electric power system to the storage electric power.

seas:incomingElectricPower Links an electric power system, or an electrical connection point, to the incoming electric power. (a) the sum of the electric power that enter the system from the exterior, or (b) the electric power that enters the system through this connection point.

seas:outgoingElectricPower Links an electric power system, or an electrical connection point, to the outgoing electric power. (a) the sum of the electric power that leaves the system from the exterior, or (b) the electric power that leaves the system through this connection point.

seas:electricPowerBalance Links an electric power system, or an electrical connection point, to the balance between the incoming electric power and the outgoing electric power.

Link to property seas:AlternatingCurrentPowerProperty**.** The class of electric power properties for alternating current. An alternating current power property cannot be directly quantified. One must quantify its individual power properties."

seas:power Links an alternating current power property to its active part (i.e., the real part of the complex value).

seas:reactivePower Links an alternating current power property to its reactive part (i.e., the imaginary part of the complex value).

seas:apparentPower Links an alternating current power property to its apparent part (i.e., the module of the complex value).

seas:powerFactor Links an alternating current power property to its power factor (i.e., the cosinus of the voltage with respect to current, unitless).

seas:importedActiveEnergy Links an alternating current power property to its imported active energy, often in Wh/10.

Link to property seas:ElectricEnergyProperty**.** The class of properties that are quantifiable and have a quantity kind of electric energy.

seas:electricEnergy Links a feature of interest to one of its electric energy properties. There may be many energy properties for a feature of interest, so this property is not functional.

seas:consumedElectricEnergy Links an electric power system to the consumed electric energy.

seas:producedElectricEnergy Links an electric power system to the produced electric energy.

seas:storageElectricEnergy Links an electric power system to the storage electric energy.

seas:incomingElectricEnergy Links an electric power system, or an electrical connection point, to the incoming electric energy. (a) the sum of the electric energy that enter the system from the exterior, or (b) the electric energy that enters the system through this connection point.

seas:outgoingElectricEnergy Links an electric power system, or an electrical connection point, to the outgoing electric energy. (a) the sum of the electric energy that leaves the system from the exterior, or (b) the electric energy that leaves the system through this connection point.

seas:electricEnergyBalance Links an electric power system, or an electrical connection point, to the balance between the incoming electric energy and the outgoing electric energy.

Classification of voltage and their evaluations .

seas:ExtraLowVoltage The voltage is Extra-Low according to IEC 60038.

seas:LowVoltage The voltage is Low according to IEC 60038.

seas:MediumVoltage The voltage is Medium according to IEC 60038.

seas:HighVoltage The voltage is High according to IEC 60038.

seas:ExtraHighVoltage The voltage is Extra-High according to IEC 60038.

seas:ExtraLowVoltageEvaluation The voltage is Extra-Low according to IEC 60038.

seas:LowVoltageEvaluation The voltage is Low according to IEC 60038.

seas:MediumVoltageEvaluation The voltage is Medium according to IEC 60038.

seas:HighVoltageEvaluation The voltage is High according to IEC 60038.

seas:ExtraHighVoltageEvaluation The voltage is Extra-High according to IEC 60038.

Link to property seas:ResistanceProperty**.** The class of properties that are quantifiable and use a quantity dimension of electric resistance.

seas:electricResistance Links an electrical connection point to one of its resistance properties. There may be many resistance properties of an electrical connection point, so this property is not functional.

seas:positiveToMidResistance Links a direct current connection point to the resistance between its positive and its mid wires.

seas:positiveToNegativeResistance Links a direct current connection point to the resistance between its positive and its negative wires.

seas:midToNegativeResistance Links a direct current connection point to the resistance between its mid and its negative wires.

seas:phaseToNeutralResistance Links a single phase connection point to the resistance between its phase and neutral wires.

seas:RSResistance Links a three-phase connection point to the resistance between its R and its S wires.

seas:STResistance Links a three-phase power bus to the resistance between its S and its T wires.

seas:TRResistance Links a three-phase power bus to the resistance between its T and its R wires.

seas:RNResistance Links a three-phase power bus to the resistance between its R and its N wires.

seas:SNResistance Links a three-phase power bus to the resistance between its S and its N wires.

seas:TNResistance Links a three-phase power bus to the resistance between its T and its N wires.

Link to property seas:ReactanceProperty**.** The class of properties that are quantifiable and use a quantity dimension of electric reactance.

seas:electricReactance Links an electrical connection point to one of its reactance properties. There may be many reactance properties of an electrical connection point, so this property is not functional.

seas:positiveToMidReactance Links a direct current connection point to the reactance between its positive and its mid wires.

seas:positiveToNegativeReactance Links a direct current connection point to the reactance between its positive and its negative wires.

seas:midToNegativeReactance Links a direct current connection point to the reactance between its mid and its negative wires.

seas:phaseToNeutralReactance Links a single phase connection point to the reactance between its phase and neutral wires.

seas:RSReactance Links a three-phase connection point to the reactance between its R and its S wires.

seas:STReactance Links a three-phase power bus to the reactance between its S and its T wires.

seas:TRReactance Links a three-phase power bus to the reactance between its T and its R wires.

seas:RNReactance Links a three-phase power bus to the reactance between its R and its N wires.

seas:SNReactance Links a three-phase power bus to the reactance between its S and its N wires.

seas:TNReactance Links a three-phase power bus to the reactance between its T and its N wires.

Link to property seas:ConductanceProperty**.** The class of properties that are quantifiable and use a quantity dimension of electric conductance.

seas:electricConductance Links an electrical connection point to one of its conductance properties. There may be many conductance properties of an electrical connection point, so this property is not functional.

seas:positiveToMidConductance Links a direct current connection point to the conductance between its positive and its mid wires.

seas:positiveToNegativeConductance Links a direct current connection point to the conductance between its positive and its negative wires.

seas:midToNegativeConductance Links a direct current connection point to the conductance between its mid and its negative wires.

seas:phaseToNeutralConductance Links a single phase connection point to the conductance between its phase and neutral wires.

seas:RSConductance Links a three-phase connection point to the conductance between its R and its S wires.

seas:STConductance Links a three-phase power bus to the conductance between its S and its T wires.

seas:TRConductance Links a three-phase power bus to the conductance between its T and its R wires.

seas:RNConductance Links a three-phase power bus to the conductance between its R and its N wires.

seas:SNConductance Links a three-phase power bus to the conductance between its S and its N wires.

seas:TNConductance Links a three-phase power bus to the conductance between its T and its N wires.

Link to property seas:SusceptanceProperty**.** The class of properties that are quantifiable and use a quantity dimension of electric susceptance.

seas:electricSusceptance Links an electrical connection point to one of its susceptance properties. There may be many susceptance properties of an electrical connection point, so this property is not functional.

seas:positiveToMidSusceptance Links a direct current connection point to the susceptance between its positive and its mid wires.

seas:positiveToNegativeSusceptance Links a direct current connection point to the susceptance between its positive and its negative wires.

seas:midToNegativeSusceptance Links a direct current connection point to the susceptance between its mid and its negative wires.

seas:phaseToNeutralSusceptance Links a single phase connection point to the susceptance between its phase and neutral wires.

seas:RSSusceptance Links a three-phase connection point to the susceptance between its R and its S wires.

seas:STSusceptance Links a three-phase power bus to the susceptance between its S and its T wires.

seas:TRSusceptance Links a three-phase power bus to the susceptance between its T and its R wires.

seas:RNSusceptance Links a three-phase power bus to the susceptance between its R and its N wires.

seas:SNSusceptance Links a three-phase power bus to the susceptance between its S and its N wires.

seas:TNSusceptance Links a three-phase power bus to the susceptance between its T and its N wires.

4.5.4 **Module** ElectricVehicleOntology

IRI: https://w3id.org/seas/ElectricVehicleOntology
Current version IRI: https://w3id.org/seas/ElectricVehicleOntology-1.0

seas:ElectricVehicleOntology imports seas:ElectricPowerSystemOntology.

Sub-classes of seas:ElectricPowerSystem

seas:ElectricVehicle**.** The class of electric vehicles. Electric vehicles can be charged using a connection to an electric vehicle service equipment.

seas:ElectricVehicleChargingStationPool**.** The class of electric vehicle charging station pools. An electric vehicle charging station pool is connected to the grid and composed of one or more electric vehicle charging stations.

seas:ElectricVehicleChargingStation**.** The class of electric vehicle charging station pools. An electric vehicle charging station pool is connected to the grid and composed of one or more electric vehicle charging stations. Est un sous-electric power system d'un pool.

seas:ElectricVehicleServiceEquipment**.** The class of Electric Vehicle Service Equipments.

Connection points

seas:CHAdeMO**.** CHAdeMO connection point.

seas:T2ConnectionPoint**.** The area of the zone, zone connection, or zone frontier.

seas:T2S**.** Type 2 with shutter connection point.

seas:T3**.** Type 3 connection point.

4.5.5 Module BatteryOntology

IRI: https://w3id.org/seas/BatteryOntology
Current version IRI: https://w3id.org/seas/BatteryOntology-1.0

seas:BatteryOntology defines batteries and their state of charge ratio property

seas:Battery The class of batteries, which are electric power storage systems.

seas:stateOfChargeRatio The state of charge of a battery, quantified by unitless values.

4.5.6 Module PhotovoltaicOntology

IRI: https://w3id.org/seas/PhotovoltaicOntology
Current version IRI: https://w3id.org/seas/PhotovoltaicOntology-1.0

seas:PhotovoltaicOntology imports seas:ElectricPowerSystemOntology and enables to describe photovoltaic-related systems.

Sub-classes of seas:ElectricPowerProducer

seas:SolarArray. Solar arrays consist of one or many solar panels.

seas:SolarPanel. Solar panels consist of one or many solar modules.

seas:SolarModule. Solar modules consist of one or many solar cells.

seas:SolarCell. Solar cells belong a solar module.

Sub-classes of seas:ElectricPowerTransformer

seas:PowerInverter. The class of electronic devices or circuitry that change direct current (DC) to alternating current (AC). (source: The Authoritative Dictionary of IEEE Standards Terms, Seventh Edition, IEEE Press, 2000, ISBN 0-7381-2601-2, page 588)

Sub-classes of seas:Actuator

seas:SolarTracker. A solar tracker is a device that orients a payload toward the Sun. Payloads are usually solar panels, parabolic troughs, fresnel reflectors, mirrors or lenses. (source: Wikipedia).

4.6 Vertical modules for the Smart Home domain

4.6.1 Module ZoneOntology

IRI: https://w3id.org/seas/ZoneOntology
Current version IRI: https://w3id.org/seas/ZoneOntology-1.0

seas:ZoneOntology imports seas:SystemOntology and seas:DeviceOntology. Zones are modeled using class seas:Zone, are connected by seas:ZoneConnection to their seas:zoneFrontier. This module defines some common properties for zones.

Systems, connections, connection points

seas:Zone. A part or a subsection of a building, campus, town, etc.

seas:subZone. Links an zone to another it is contained in.

seas:ZoneConnection. When they are connected, zones may exchange light, heat, humidity, agents.

seas:ZoneFrontier. Surface that marks the frontier of an zone, and represents the connection point to other zones.

Functional sub-properties of seas:hasProperty

seas:volume. Quantifies the volume of the zone.

seas:area. The area of the zone, zone connection, or zone frontier.

seas:humidity. Links a zone, zone connection, or zone frontier, to the property that qualifies its humidity. The humidity property has itself properties that quantify it, such as the specific humidity.

- seas:HumidityProperty. The class of humidity properties of zones, zone connections, or zone frontiers
- seas:specificHumidity. The specific humidity is defined as the mass of water per unit mass of moist air.
- seas:absoluteHumidity. The absolute humidity is defined by the mass of water vapor per humid air volume.
- seas:saturatedVapourPressure. The saturated vapor pressure of moist air.

seas:population. Links a zone, zone connection or zone frontier to the property that quantifies its population in terms of agents. The population property may itself have properties such as the population flow.

- seas:PopulationProperty. The class of population properties of zones, zone connections, or zone frontiers.

seas:populationFlow. The flow of agents that cross a zone, a zone connection, or a zone frontier.

Sensors and actuators For now, module seas:ZoneOntology defines the following sensors:

seas:HumiditySensor. Humidity sensors observe the humidity of a zone, zone connection, or zone frontier.

seas:HumidityActuator. An actuator that is capable of modifying the humidity in zones.

seas:PresenceSensor. A Presence sensor observes the population inside a zone, zone connection or zone frontier.

4.6.2 Module ThermodynamicSystemOntology

IRI: https://w3id.org/seas/ThermodynamicSystemOntology
Current version IRI: https://w3id.org/seas/ThermodynamicSystemOntology-1.0

seas:ThermodynamicSystemOntology imports seas:SystemOntology, and describes thermodynamic systems that exchange heat, and their properties.

Systems, connections, connection points

seas:ThermodynamicSystem. The class of systems that produce, dissipate, and exchange thermal energy.

seas:subThermodynamicSystemOf. Links a thermodynamic system to another whose internal energy it contributes to.

seas:exchangesHeatWith. Links two thermodynamic systems between which heat flows.

Functional sub-properties of seas:hasProperty

seas:temperature. The temperature of a thermodynamic system.

seas:thermalTransmittance. Thermal Transmittance of the thermodynamic system.

seas:totalHeatTransfer. The total heat transfer.

4.6.3 Module ZoneLightingOntology

IRI: https://w3id.org/seas/ZoneLightingOntology
Current version IRI: https://w3id.org/seas/ZoneLightingOntology-1.0

seas:ZoneLightingOntology imports seas:SystemOntology and seas:DeviceOntology, and describes lighting sources that illuminate zones.

The SEAS Zone Lighting ontology describes light sources and illuminable zones. Light sources and light source devices (subclass of seas:Device) may seas:illuminates illuminable zones (subclass of seas:Zones), while illuminable systems may transmit light to other illuminable zones (property seas:transmitsLightTo)

Main source is http://www.compuphase.com/electronics/candela_lumen.htm.

Systems, connections, connection points

seas:IlluminableZone. The class of zones _in_ which one may measure/effect luminosity, and perceive brightness. Illuminable zones include the exterior, rooms, skylight systems. There is no particular assumption about how the light is transmitted in an illuminable zone. Meaning: the luminosity may be non homogeneous in an illuminable zone.

seas:transmitsLightTo. The subject illuminable zone transmits light to the object illuminable zone. This property may be qualified by class seas:LightTransmissionSystemConnection. To account for one-way mirrors, this property is not symmetric.

seas:LightTransmissionSystemConnection. The class of connections between illuminable zones, through which light can be transmitted. Examples of light transmission system connections include glasses, tinted glasses, one-way mirrors.

seas:IlluminableZoneFrontier. The class of zones frontiers _on_ which one may measure/effect luminosity, and perceive brightness. Illuminable zones are surfaces such as walls, tables, sheer curtains, mirrors, window. Light may be reflected, absorbed, and transmitted by illuminable zone frontiers.

Functional sub-properties of seas:hasProperty

seas:refractiveIndice. Qualifies the refractive indice of an illuminable zone.

seas:lightTransmission. The light transmission through an illuminable zone frontier.

seas:lightReflection. The light reflection at an illuminable zone frontier.

seas:luminosity. The luminosity at an illuminable zone frontier.

seas:colour. The colour of a light source.

Light sources

seas:LightSource. The class of systems that are capable of producing light.

seas:LightActuator. The class of devices that are capable of producing light.

seas:subLightSourceOf. Links an light source to another light source that contributes to it.

seas:illuminates. The subject light system may illuminate the object light system.

Light sensors

seas:LightSensor. The class of devices that are capable of generating evaluations of the luminosity in a zone.

4.6.4 **Module** BuildingOntology

IRI: https://w3id.org/seas/BuildingOntology
Current version IRI: https://w3id.org/seas/BuildingOntology-1.0

seas:BuildingOntology imports the ontology seas:ZoneOntology. This module describes a taxonomy of buildings, building spaces, and rooms. Some categorizations are based on the energy efficiency related to their insulation etc., although the actual values for classes depend the country specific regulations and geographical locations. Other categorizations are based on occupancy and activities. There is no single accepted categorization available. This taxonomy uses some types selected from:

- International building occupancy based categories (USA)
- The Classification of Types of Constructions (EU)
- Finnish building categorization VTJ2000 (Finland)
- Wikipedia category page for Rooms: https://en.wikipedia.org/wiki/Category:Rooms

This ontology should limit the overlap with the following existing specialized ontologies:

- FIEMSER ontology.
- gbXML ontology.
- ifcOWL ontology, which is a OWL version of the Building Information Model (BIM) standard.
- the potential ontology that might be developed in the context of the Linked Building Data community group, or its successor Working Group.

Hierarchy of sub-classes of seas:Zone

seas:SiteOfBuilding Building site is a locale containing one or more separate buildings. They are zones.

seas:Construction Constructions are structures connected with the ground which are made of construction materials and components and/or for which construction work is carried out. They are zones.

seas:CivilEngineeringWork Civil engineering works are all constructions not classified under buildings : railways, roads, bridges, highways, airport runways, dams etc.

seas:BuildingSpace A Space is a 3D volume bounded by surfaces. According to the FIEMSER definition, a building space in SAREF defines the physical spaces of the building. They are zones.

seas:BuildingSpatialStructure A man made structure with spatial properties.

seas:BuildingStorey The storey represents a (nearly) horizontal aggregation of spaces that are vertically bound.

seas:Building Buildings are roofed constructions which can be used separately, have been built for permanent purposes, can be entered by persons and are suitable or intended for protecting persons, animals or objects. They are building spaces, constructions, and zones.

seas:NormHouse A building fulfilling the minimal criteria for energy efficiency.

seas:LowEnergyHouse A house typically consuming half the energy than a norm house.

seas:PassiveHouse A house typically consuming a quarter of the energy than a norm house.

seas:ZeroEnergyBuilding A net zero-energy building (ZEB) is a building that over a year does not use more energy than it generates.

seas:PlusEnergyBuilding A net plus-energy building is a building that over a year does generates more energy than it uses.

seas:ResidentalBuilding A residential building is a building at least half of which is used for residential purposes.

seas:HolidayBuilding A secondary residential building used only occasionally during vacations such as a summerhouse or cottage.

seas:SmallHouse A detached small residential building.

seas:OneDwellingBuilding Detached house.

seas:NonResidentalBuilding A non-residential building is a building at least half of which is used for other than residential purposes.

seas:MercantileBuilding Places where goods are displayed and sold. Examples: grocery stores, department stores, and gas stations.

seas:OfficeBuilding Places where services are provided. Examples: banks, insurance agencies.

seas:AssemblyBuilding places used for people gathering for entertainment, worship, and eating or drinking. Examples: churches, restaurants.

seas:InstitutionalBuilding Institutions such as hospitals providing medical and surgical treatment and nursing care for ill or injured people.

seas:EducationalBuilding Schools and day care centers.

seas:IndustrialBuilding Buildings used for industrial production, e.g. factories, workshops, slaughterhouses, breweries, assembly plants, etc.

seas:PowerplantBuilding Places housing any type of a power plant.

seas:StorageBuilding Places where items are stored. Examples: warehouses, reservoirs and silos.

Sub-classes of seas:BuildingSpace

seas:Balcony An accessible structure extending from a building, especially outside a window.

seas:Stairs A construction designed to bridge a large vertical distance by dividing it into smaller vertical distances, called steps.

seas:Yard A small usually walled and often paved zone open to the sky and adjacent to a building.

seas:Greenhouse A building, room, or zone, usually chiefly of glass, in which the temperature is maintained within a desired range, used for cultivating tender plants or growing plants out of season.

seas:TreeHouse A small house, especially one for children to play in, built or placed up in the branches of a tree.

seas:Lobby An entrance hall, corridor, or vestibule, as in a public building, often serving as an anteroom; foyer.

- seas:Hall A large entrance room of a house or building.
- seas:Corridor A gallery or passage connecting parts of a building; hallway.

seas:Room A room in a building space enclosed by surfaces, this could also be modelled as role of space, not subclass of the space itself.

- seas:LivingRoom Living Room is the main room of daytime activity.
- seas:Kitchen Kitchen is a room used mainly for cooking and possibly eating.
- seas:Bedroom Bedroom is used mainly for sleeping.
- seas:Elevator Elevator is used to transport people between different floors.
- seas:Bathroom Bathroom is mainly used for bathing & washing up related activities.
- seas:Sauna Sauna is a special type bathroom for enjoying heated steam.
- seas:StorageRoom Room for storage.
- seas:Garage Room for garage.
- seas:UtilityRoom Room for other special utilities and hobbies.
- seas:DiningRoom A room in which meals are eaten, as in a home or hotel, especially the room in which the major or more formal meals are eaten.
- seas:Office A room, set of rooms, or building where the business of a commercial or industrial organization or of a professional person is conducted.
  - seas:HomeOffice A work or office space set up in a person's home and used exclusively for business on a regular basis.
- seas:Basement A story of a building, partly or wholly underground.
- seas:Laundry A room or zone, as in a home or apartment building, reserved for doing the family wash.
- seas:Attic the part of a building, especially of a house, directly under a roof; garret.
- seas:SwimmingPool A tank or large artificial basin, as of concrete, for filling with water for swimming.
- seas:Sunroom A structure, either attached or integrated into a building, which allows enjoyment of the surrounding landscape while being sheltered from adverse weather.

Sub-classes of seas:ZoneConnection

seas:BuildingSpaceConnection Connection between two building spaces, where they may potentially exchange heat, humidity, agents.

- seas:Roof A (tilted more than 60 percent) vertical surface that separates building spaces.
- seas:Wall A roof upwards tilted surface tilted up part of upper envelope of building.
- seas:Door the large flat piece of wood, glass etc that you move when you go into or out of a building, room, vehicle etc, or when you open a cupboard (Longman Dictionary of Contemporary English Online)
- seas:Window a space or an zone of glass in the wall of a building or vehicle that lets in light (Longman Dictionary of Contemporary English Online)
- seas:ClosedBuildingSpaceConnection A closed building space connection is a closed connection between two building spaces. This separation may for instance be a wall, i.e., a (tilted more than 60 percent) vertical surface. Instances of seas:ClosedBuildingSpaceConnection may be typed by classes from specialized building ontologies.
- seas:OpennableBuildingSpaceConnection An opennable building space connection is a connection between two building spaces that has a certain degree of openness. This connection may for instance be a window or a door. Instances of seas:OpennableBuildingSpaceConnection may be typed by classes from specialized building ontologies.
- seas:OpenBuildingSpaceConnection An open building space connection is a connection between two building spaces that is open.This connection may for instance be a hole in a wall, or a virtual separation between two offices in an open space. Instances of seas:OpenBuildingSpaceConnection may be typed by classes from specialized building ontologies.

Sub-classes of seas:ZoneFrontier

seas:BuildingSpaceFrontier Surface that marks the frontier of a building space, and represents the connection point to other building spaces.

- seas:Ceiling Ceiling is a downwards tilted horizontal surface.
- seas:Floor A floor is a upwards tilted horizontal surface, could be divided to interior, exposed (outside) or raised floor.

4.7 Vertical modules for the SEAS architecture

4.7.1 Module ArchitectureOntology

IRI: https://w3id.org/seas/ArchitectureOntology
Current version IRI: https://w3id.org/seas/ArchitectureOntology-2.0

seas:ArchitectureOntology imports the ontology seas:SystemOntology. This module defines systems and their connections for the SEAS IT architecture.

Systems

seas:Entity. The class SEAS entity represents the abstract class of communicating nodes.

seas:CoreEntity. SEAS core entities are SEAS nodes that belong to the SEAS core domain. The SEAS core domain is where all SEAS core entities, group managers, and SEAS core services can be interconnected. Interconnections within the SEAS core domain all rely on the Internet Protocol. SEAS core entities can be connected to other SEAS core entities in a peer-to-peer basis, and exchange information. SEAS core entities can be connected to SEAS field entities, provided that they have sufficient access right.

seas:RegistrationEntity. The class of entities that provide the SEAS core registration services for SEAS core entities. A SEAS registration service is an API that enables SEAS core services to register themselves along with their capabilities for others to find them.

seas:TransactionEntity. The class of entities that provide the SEAS core transaction services for SEAS core entities. A SEAS core transaction service is an API that connects SEAS core entities to banks and payment systems, and acts as a trusted notary to enable compensation for participation in SEAS field entities management.

seas:OntologyEntity. The class of entities that provide the SEAS core ontology services for SEAS core entities. A SEAS core ontology service is an API that exposes media types descriptions, notably information about:

- how to validate a document with a specific media type;
- how to lift a document with a specific media type to RDF;
- how to validate a RDF Graph;
- how to lower a RDF Graph to a document with a specific media type.

seas:SecurityEntity. The class of entities that provide the SEAS core security services for SEAS core entities. A SEAS core security service is an API that enable:

- SEAS core entities to communicate their access control preferences;
- SEAS core entities to control the access rights of other SEAS core entities for some type information.

seas:EndUser. The class of end user entities, that own SEAS field entities.

seas:FieldEntity. SEAS field entities are SEAS nodes that belong to the SEAS field domain. The SEAS field domain is the link with the physical world. SEAS field entities can be connected to other SEAS field entities in a client-server basis. SEAS field entities can belong to one or several SEAS groups. Every SEAS group is managed by exactly one group manager. Field entities participate in the management of the group.

seas:GroupManager. SEAS group managers are both field and core entities. They manage exactly one group.

seas:Gateway. SEAS Gateways are group managers equipped for interfacing with entities in a field that use different communication protocols.

Connections

seas:EntityConnection. The sub-class of connections that relate SEAS entities.

seas:exchangesInformationWith Links a SEAS core entity to another SEAS core entity it exchanges information with. For example, for information to flow between a system operator and a market operator, then they must share a core entity connection:

<system_operator> a seas:CoreEntity .
<market_operator> a seas:CoreEntity .
<system_operator> seas:exchangesInformationWith <market_operator> .

If there is a core entity connection between several core entities, then one may infer these core entities can exchange information. This property can be qualified using class seas:CoreEntityConnection, which connects (at least) the two core entities.

seas:EntityConnection. The class of SEAS core entity connections between SEAS core entities. SEAS core entity connections describe potential information flows between the SEAS core entities it connects. For example, the following RDF graph states that a SEAS core entity connection connects a home management system with an aggregator and the transaction server:

<connection> a seas:ElectricalConnection ;
seas:connectsSystem <home_management_system> , <aggregator> , <transaction_server> .

Only SEAS core entities are connected through a SEAS core entity connection. A connection between SEAS core entities is not necessarily a SEAS core entity connection.

seas:directlyAccesses. Links a SEAS core entity to a SEAS field entity it has direct access to. For example, for a flexibility operator to directly access a solar panel, then it must have a direct access connection to it:

<flexibility_operator> a seas:CoreEntity .
<solar_panel> a seas:FieldEntity .
<flexibility_operator> seas:accessesDirectly <solar_panel> .

If there is a direct access connection between a core entity and a field entity, then one may infer the core entity can directly access the field entity.

La propriété inverse de seas:directlyAccesses est seas:directlyAccessedBy.

This property can be qualified using class seas:DirectAccessConnection, which connects the core entity and the field entity. The class of SEAS direct access connections between a SEAS core entity and one or more SEAS field entities it directly accesses.

seas:DirectAccessConnection. Direct access connections describe potential control of the SEAS core entity to the SEAS field entities. For example, the following RDF graph states that a SEAS direct access connection connects a distribution system operator with a solar panel:

<connection> a seas:DirectAccessConnection ;
seas:connectsSystem <distribution_system_operator> , <solar_panel> .

Only SEAS entities are connected through a direct access connection. A connection between SEAS entities is not necessarily a direct access connection.

seas:directlyManages. Links a SEAS group manager to a SEAS field entity it directly manages. For example, for a home management system to directly manage a smart fridge, then it must have a management connection to it:

<home_management_system> a seas:GroupManager .
<smart_fridge> a seas:FieldEntity .
<home_management_system> seas:directlyManages <smart_fridge> .

If there is a direct management connection between a group manager and a field entity, then one may infer

the group manager directly manages the field entity.

The inverse property of seas:directlyManages is seas:directlyManagesBy.

This property can be qualified using class seas:DirectManagementConnection, which connects the group manager and the field entities it directly manages.

seas:DirectManagementConnection. The class of SEAS direct management connections between a SEAS group manager entity and each of the entities it directly manages (i.e., the group). Direct management connections describe the management of the SEAS field entities by the SEAS group manager. For example, the following RDF graph states that a home management system manages a group of field entities composed of a solar panel and a battery.

<connection> a seas:DirectManagementConnection ;
seas:connectsSystem <home_management_system> , <solar_panel> , <battery> .

Only SEAS entities and SEAS field entities are connected through a direct management connection. A connection between SEAS entities is not necessarily a direct management connection.

Connection points

seas:EntityAccessPoint. The class of access points of SEAS entities, at which they may access or be accessed by other entities. For example, a solar panel may have an access point where smart grid management systems can access the latest information about its electricity production. An entity access point describes the APIs that one may use to access the system. An entity access point only belongs to one entity, and connects it through entity connections. A connection point of an entity is not necessarily an entity access point.

4.8 Vertical modules for the offers and the markets

In the case of offers and markets, we directly import the GoodRelations ontology, and specialize some of its core classes.

4.8.1 Module OfferingOntology

IRI: https://w3id.org/seas/OfferingOntology
Current version IRI: https://w3id.org/seas/OfferingOntology-1.0

seas:OfferingOntology imports the ontologies Process Execution Platform ontology and seas:PlayerOntology.

Offers/demands are modeled by a link seas:offers or seas:seeks between a seas:Player and a seas:Offering, which is a sub-class of pep:Process. Conditions of the offer or the demand are described using property pep:hasOutput. The offer can be annotated with a temporal validity context using property seas:hasTemporalContext.

The agreement between two business players is modeled using class seas:Contract, which is a sub-class of pep:ProcessExecutor, and links business players using property seas:party. The process that the contract executions implement model what the players agreed upon.

The contract execution is modeled by class seas:Transaction, that is a sub-class of pep:ProcessExecution.

Figure 9 illustrates classes and properties of module OfferingOntology.

seas:Offering. An offering is a process that is offered or sought for by a player. An offering may be linked through property seas:includes to the property evaluations that are included in the offering. For example, <player1> offers to pay 34 to receive 1 MWh.

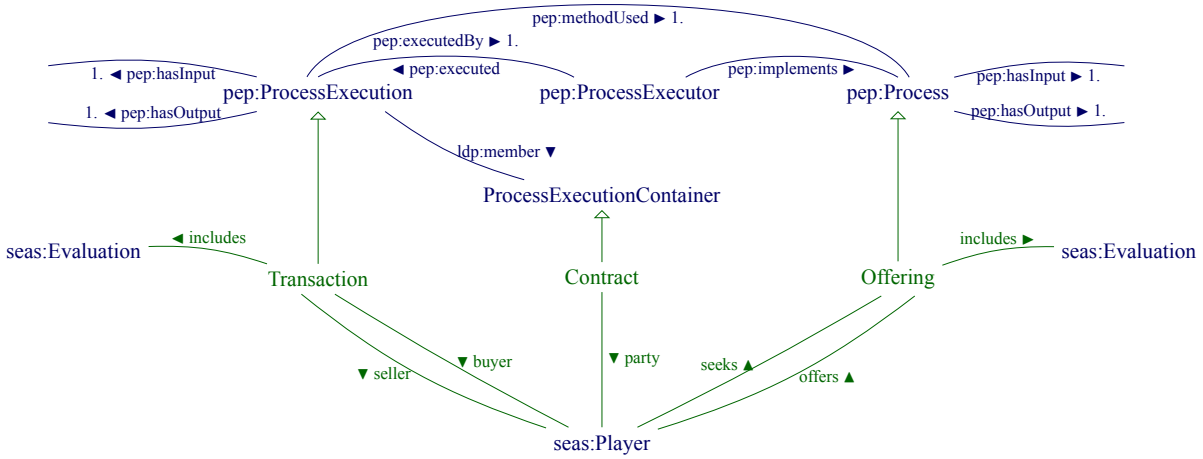

Figure 12: Module seas:OfferingOntology. Prefix ldp stands for http://www.w3.org/ns/ldp#, see the Linked Data Platform [16].

```
<player1> seas:moneyConnectionPoint <connectionPoint1> ;
  seas:electricalConnectionPoint <connectionPoint2> ;
  seas:offers <offering> .

<connectionPoint1> seas:moneyBalance <property1> .
<connectionPoint2> seas:energyBalance <property2> .
<property1> seas:evaluation <evaluation1> .
<property2> seas:evaluation <evaluation2> .

<offering> seas:includes <evaluation1>, <evaluation2> .

<evaluation1> seas:evaluatedValue "-34"^^cdt:euro .
<evaluation2> seas:evaluatedValue "1 MWh"^^cdt:ucum .
```

seas:offers. Links an agent to a process it offers.

seas:seeks. Links an agent to a process it seeks.

seas:Contract. A contract is a container for transactions.

seas:party. Links a contract to one of the parties that agreed on the contract.

seas:Transaction. A transaction is a contract execution. A transaction has a buyer, a seller, and may include zero or more evaluations. The transaction content is modeled by the included evaluations. In particular, the transaction price is modeled as the evaluation of a money flow between the buyer and the seller.

seas:buyer. Links a transaction to the buyer.

seas:seller. Links a transaction to the seller.

seas:includes. Links a transaction, contract or offering to the property evaluations it includes. More general transactions, contracts, or offerings can be described using property pep:hasOutput.

4.8.2 **Module** TradingOntology

IRI: https://w3id.org/seas/TradingOntology
Current version IRI: https://w3id.org/seas/TradingOntology-1.0

seas:TradingOntology définit:

- players own systems and trade commodities, which have a price;
- bilateral electricity contracts are connections between electricity traders at which they exchange electricity;
- electricity markets are connections between electricity traders at which they exchange electricity, using a market license;
- electricity markets can be cleared, and balanced;
- evaluations can have a traded volume validity context.

Ownership, commodities and trading

seas:owns Links a player to a system it owns.

seas:isOwnedBy Links a system to the one and only player that owns it.

seas:trades Links a player to a commodity it trades.

seas:isTradedBy Links a commodity to the one and only player that trades it.

seas:Commodity In economics, a commodity is a marketable item produced to satisfy wants or needs. Often the item is fungible. Economic commodities comprise goods and services (source: Wikipedia).

seas:Good In economics, a good is a material that satisfies human wants and provides utility, for example, to a consumer making a purchase while getting an enough-satisfying product. A common distinction is made between *goods* that are tangible property, and services, which are non-physical (source: Wikipedia).

seas:Service In economics, a service is an economic activity where an immaterial exchange of value occurs. When a service such as labor is performed the buyer does not take exclusive ownership of that which is purchased, unless agreed upon by buyer and seller. The benefits of such a service, if priced, are held to be self-evident in the buyer's willingness to pay for it (source: Wikipedia).

seas:AncillaryService Ancillary services are the specialty services and functions provided by the electric grid that facilitate and support the continuous flow of electricity so that supply will continually meet demand. (source: Wikipedia). These include black start capability (the ability to restart a grid following a blackout); frequency response (to maintain system frequency with automatic and very fast responses); fast reserve (which can provide additional energy when needed); the provision of reactive power and various other services. (source: Entso-e)

Prices

seas:PriceProperty The class of price properties, whose value is expressed using some currency.

seas:NettPriceProperty The class of nett price properties, that have no tax.

seas:GrossPriceProperty The class of gross price properties, that have some tax. A gross price has some properties such as a nett price, and some taxes.

seas:price The price of some commodity.

seas:grossPrice Links a price property to its gross price, including any taxes.

seas:nettPrice Tax for the commodity, quantified as a percentage of something. Functional sub properties of seas:hasTax must precise what that *something* is.

seas:tax Tax for the commodity, quantified as a percentage of something. Functional sub properties of seas:hasTax must precise what that *something* is.

seas:valueAddedTax A value-added tax (VAT), known in some countries as a goods and services tax (GST), is a type of general consumption tax that is collected incrementally, based on the value added, at each stage of production and is usually implemented as a destination-based tax, where the tax rate is based on the location of the customer. (source: Wikipedia). The value added tax is quantified as a percentage of the nett price.

seas:operatedBy Links some connection to a player that regulates what's happening through that connection.

Bilateral contracts are Connections

seas:tradesElectricityWith An Electricity Player can trade electricity directly with another electricity player, through an electricity bilateral contract. They then exchange some of the commodities they trade. These connections are qualified with class seas:ElectricityBilateralContract.

seas:ElectricityBilateralContract An electricity bilateral contract is both a connection between electricity players, and a contract. It only connects electricity traders. It is linked to some player that validates each trade: the operator.

Markets are Connections

seas:tradesElectricityOn An Electricity Player can trade electricity on an Electricity Market. These connections are qualified with class seas:ElectricityMarketLicense.

seas:ElectricityMarketLicense An Electricity Market License is a connection point between a player and a market.

seas:hasElectricityMarketLicense Links an electricity player to its electricity market license.

For example,

```
<http://market.com/> a seas:ElectricityMarket .
  seas:connectsSystem <http://agregator.com> , <http://otheragregator.com> .
  seas:connectsSystemAt <http://market.com/license12> , <http://market.com/license32> .

<http://agregator.com> a seas:ElectricityTrader ;
  seas:tradesElectricityOn <http://market.com/> ;
  seas:hasElectricityMarketLicense <http://market.com/license12> .

<http://otheragregator.com> a seas:ElectricityTrader ;
  seas:tradesElectricityOn <http://market.com/> ;
  seas:hasElectricityMarketLicense <http://market.com/license32> .
```

seas:Clearing In economics, market clearing is the process by which, in an economic market, the supply of whatever is traded is equated to the demand, so that there is no leftover supply or demand (source: Wikipedia).

seas:ClearingExecution The execution of some clearing process on an electricity market.

seas:Balancing Balancing refers to the process in which a Balance responsible party acts to ensure that demand is equal to supply after markets have closed, in and near real time (source: Entso-e).

seas:BalancingExecution The execution of some balancing process on a market.

Traded volume context

seas:hasTradedVolumeContext Links an evaluation to its traded volume validity context.

4.9 Using or Contributing to the Ontologies

The concepts in the SEAS ontologies can be used in RDF Graphs, can be exposed as the content of a RDF Source on the Web, or sent to a Web server or client in a message.

There are two recommended ways to browse the SEAS ontologies:

- start the Protégé ontology editor, then File → Open from URL: https://w3id.org/seas/;
- browse the Website https://w3id.org/seas/.

One can also search for some text in the search field at the following URL: https://github.com/thesmartenergy/seas/.

As the SEAS ontologies are open-source on GitHub, contributions can be proposed after creating a GitHub account, then:

- create an issue on GitHub: https://github.com/thesmartenergy/seas/issues;
- fork the project, make your changes in a separate branch, and issue a pull request from this branch to the upstream branch.

5 Conclusion

The Knowledge Model can hence enable to describe information from different sources. Once integrated, the information from these different sources can be augmented using algorithms from work packages 3 and 4, augmenting in the same time our understanding of the energy consumption of a grid, or predict spikes in the consumption. This is part of the SEAS initial view concept, illustrated on Figure 1 below.

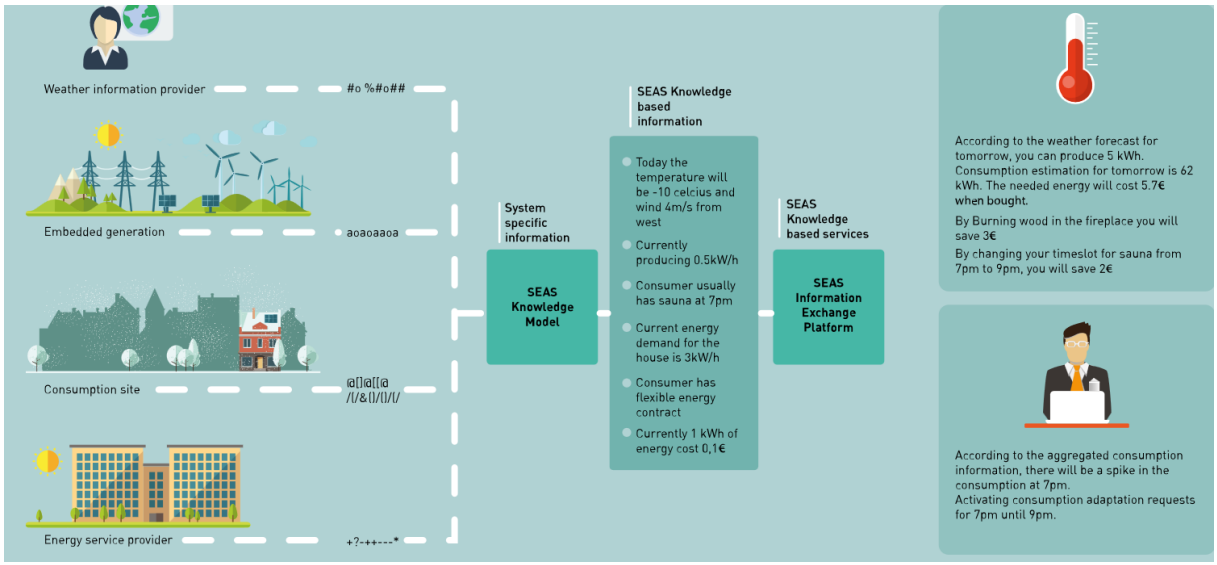

Figure 13: SEAS initial view

The SEAS Knowledge Model consists of an innovative Web ontology that is designed to: (i) meet the current best practices in terms of quality, metadata, and publication, (ii) reuse or align to existing standards, and (iii) cover the required expressivity for the SEAS use cases, while being extensible to other use cases and domains (gas, water, air, waste management).
The current version of the SEAS Knowledge Model is available at URL https://w3id.org/seas/, which is also the namespace of every concept it defines. As of November 25th, it contains a total of 33 modules, 4174 axioms, 294 classes, 216 object properties, 6 data properties, and 44 individuals. Contributions can be suggested to this model at to any open-source project at https://github.com/thesmartenergy/seas.

This work led to publications in Semantic Web conferences: [9, 7, 5], and to open-source software development reusable by third parties.

Besides the SEAS ontologies, T 2.2 partners led three pieces of work that aim atur lowering the overhead for companies, web services and constrained devices to embrace the Semantic Web formalisms and tool. More precisely, these aim at: (1) lowering the cost of prototyping RDF generation from documents having various formats. SPARQL-Generate can be used to specify how documents in heterogeneous formats can be lifted to RDF. Website https://w3id.org/sparql-generate/ contains resources to test and learn SPARQL-Generate online, or to start using it as a library in other projects.[11, 13, 12]; (2) lowering the required effort for web services and constrained devices to reach *semantic interoperability*, i.e., to get the meaning of the messages they exchange. The RDFP paves the way for a Web where things and services can discover how they can interoperate semantically. Website https://w3id.org/rdfp/ contains the RDFP ontology and its documentation, along with online demonstration of the extended Linked Data principles, and a first implementation of these principles on top of Apache Jersey [6, 14]; (3) enabling the on-the-fly support of arbitrary custom datatypes in RDF and SPARQL engines. The specification of the principles are available at website https://w3id.org/lindt/ [8, 10].

Bibliography

[1] Linked data. Published online at http://www.w3.org/DesignIssues/LinkedData.html, 2005. W3C Design issue.

[2] Tim Berners-Lee. Cool URIs don't change. W3C Note, W3C, 1998.

[3] CIM EA community. CIM EA add-in. retrieved from http://www.cimea.org/, 2015.

[4] Özden Erçin, Takoua Ghariani, Olli Huotari, Jarmo Kalaoja, and Maxime Lefrançois. SEAS Information Analysis. Deliverable D 2.1, ITEA2 12004 SEAS, 2015.

[5] Luis Gomes, Maxime Lefrançois, Pedro Faria, and Zita Vale. Publishing Real-time Microgrid Consumption Data on the Web of Linked Data. In *IEEE Power Systems Conference, (PSC 2016)*, Clemson, SC, USA, Mar 2016.

[6] Maxime Lefrançois. Interopérabilité sémantique libérale pour les services et les objets. In *Actes de la 17ème conférence Extraction et Gestion des Connaissances (EGC'17)*, Grenoble, France, January 2017.

[7] Maxime Lefrançois, Guillaume Habault, Caroline Ramondou, and Eric Françon. Outsourcing electric vehicle smart charging on the web of data. In *The First International Conference on Green Communications, Computing and Technologies*, Nice, France, July 2016.

[8] Maxime Lefrançois and Antoine Zimermann. Supporting Arbitrary Custom Datatypes in RDF and SPARQL. In *Proceedings of the Extended Semantic Web Conference, ESWC*, May 2016.

[9] Maxime Lefrançois and Antoine Zimmermann. LinkedVocabularyEditor: une extension MediaWiki pour l'édition collaborative et la publication de vocabulaires liés. In *Actes des 26e Journées Francophones d'Ingénierie des Connaissances, (IC'15)*, Rennes, France, June 2015.

[10] Maxime Lefrançois and Antoine Zimmermann. Support uniforme de types de données personnalisés dans rdf et sparql. In *Actes de la 17ème conférence Extraction et Gestion des Connaissances (EGC'17)*, Grenoble, France, January 2017.

[11] Maxime Lefrançois, Antoine Zimmermann, and Noorani Bakerally. Flexible RDF generation from RDF and heterogeneous data sources with SPARQL-Generate. In *Proceedings of the 20th International Conference on Knowledge Engineering and Knowledge Management (EKAW'16)*, Bologna, Italy, November 2016.

[12] Maxime Lefrançois, Antoine Zimmermann, and Noorani Bakerally. A SPARQL extension for generating RDF from heterogeneous formats. In *Proc. Extended Semantic Web Conference, (ESWC 2017)*, 2017.

[13] Maxime Lefrançois, Antoine Zimmermann, and Noorani Bakerally. Génération de RDF à partir de sources de données aux formats hétérogènes. In *Actes de la 17ème conférence Extraction et Gestion des Connaissances (EGC'17)*, Grenoble, France, January 2017.

[14] Maxime Lefrançois, Antoine Zimmermann, and Andrei Ciortea. Enabling flexible semantic interoperability for things and services on the Web of Data. In *Proc. Extended Semantic Web Conference, (ESWC 2016)*, 2017.

[15] Leo Sauermann and Richard Cyganiak. Cool URIs for the Semantic Web. W3C Note, W3C, December 03 2008.

[16] Steve Speicher, John Arwe, and Ashok Malhotra. Linked Data Platform 1.0. W3C Recommendation, W3C, February 26 2015.

[17] Sparks Systems. Effective ontology development using the UML and Enterpise Architect. retrieved from http://www.sparxsystems.com, 2013.

[18] Pierre-Yves Vandenbussche and Bernard Vatant. Metadata recommendations for linked open data vocabularies. Web document, 2012.

[19] W3C OWL Working Group. OWL 2 Web Ontology Language Document Overview (Second Edition), W3C Recommendation 11 December 2012. Technical report, W3C, 2012.

[20] W3C OWL Working Group. OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax (Second Edition), W3C Recommendation 11 December 2012. Technical report, W3C, 2012.

[21] Jesper Zedlitz. From UML to OWL 2. *Knowledge Technology*, 2012.