Preprint Cambridge Centre for Computational Chemical Engineering ISSN 1473 – 4273

OntoPowerSys: A Power Systems Ontology for Cross Domain Interactions in an Eco Industrial park

Aravind Devanand[1,4], Gourab Karmakar[4], Nenad Krdzavac[4],
Kevin Aditya[4], Rémy Rigo-Mariani[4], Ashok Krishnan [5], Y.S. Foo Eddy[4,5],
Iftekhar A. Karimi[1,4], Markus Kraft[2,3,4]

released: 04 June 2019

[1] Department of Chemical and
Biomolecular Engineering,
National University of Singapore,
Engineering Drive 4,
Singapore,
117585.

[2] Department of Chemical Engineering
and Biotechnology,
University of Cambridge,
New Museums Site,
Pembroke Street,
Cambridge, CB2 3RA
United Kingdom.

[3] School of Chemical and
Biomedical Engineering,
Nanyang Technological University,
62 Nanyang Drive,
Singapore,
637459.

[4] Cambridge Centre for Advanced Research
and Education in Singapore,
1 Create Way, 05-00,
Singapore,
138602.

[5] School of Electrical and
Electronic Engineering
Nanyang Technological University,
50 Nanyang Avenue,
Singapore
639798.

Preprint No. 230

UNIVERSITY OF
CAMBRIDGE

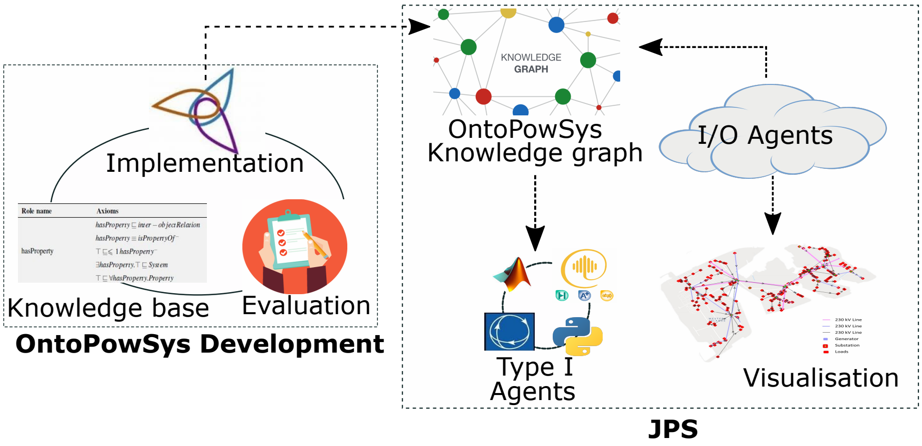
Keywords: Eco Industrial Park, ontology, power system, Optimum Power Flow

Edited by
Computational Modelling Group
Department of Chemical Engineering and Biotechnology
University of Cambridge
Philippa Fawcett Drive
Cambridge CB3 0AS
United Kingdom

E-Mail: c4e@cam.ac.uk
World Wide Web: `http://como.ceb.cam.ac.uk/`

Abstract

Knowledge management in multi-domain, heterogeneous industrial networks like an Eco Industrial Park (EIP) is a challenging task. In the present paper, an ontology based management system has been proposed for tackling this challenge. It focuses on the power systems domain and provides a framework for integrating this knowledge with other domains of an EIP. The proposed ontology, OntoPowSys is expressed using a Description Logics (DL) syntax and OWL2 language was used to make it alive. It is then used as a part of the Knowledge Management System (KMS) in a virtual EIP called the J-Park Simulator (JPS). The advantages of the proposed approach are demonstrated by conducting two case studies on the JPS. The first case study illustrates the application of Optimum Power Flow (OPF) in the electrical network of the JPS. The second case study plays an important role in understanding the cross domain interactions between chemical and electrical engineering domains in a biodiesel plant of the JPS. These case studies are available as web services on the JPS website. The results showcase the advantages of using ontologies in the development of decision support tools. These tools are capable of taking into account contextual information on top of data during their decision making process. They are also able to exchange knowledge across different domains without the need for a communication interface.



Highlights

- A domain ontology for power systems in an Eco Industrial Park.
- Implementation of the ontology as a Knowledge Mangament System of J-Park Simulator.
- Application of the ontology for the automated generation of grid topology.
- Application of the ontology for studying cross domain interactions in a biodiesel plant.

Contents

1 Introduction

Over the last few decades, public opinion has shifted towards holding industries responsible for any environmental damage they may cause. This has necessitated a shift towards a framework wherein economic growth and environmental preservation can go hand in hand [30]. One solution which has been proposed is the development of industrial clusters called Eco-Industrial Parks (EIPs). An EIP is a cluster of manufacturing and service businesses seeking enhanced environmental and economic performance. It is realised through the collaborative management of environmental and resource issues, including those related to power, water, and materials. By working in synergy, the businesses seek to achieve collective benefits which can outweigh the sum of the individual benefits [18].

The optimal operation of an EIP requires an exchange of knowledge across multiple domains and an in-depth understanding of their dynamics. Establishing seamless and effective communication between different entities in an EIP is a demanding task that requires addressing the following challenges [34]. First and foremost conundrum is the distributed nature of data storage. Furthermore, the semantics and syntax heterogeneity [7] that arise due to segregated domain silos requires consideration.

With the advent of Industry 4.0, the entities in an EIP can exchange knowledge with each other using the internet [16]. This can lead to the creation of cyber-physical systems [14] that are capable of autonomous decision making [13, 31]. The important features of Industry 4.0 are the availability and transparency of information, cross-domain application and decentralised decision making.

In the present paper, the aforementioned features of Industry 4.0 are implemented with the help of a KMS for an EIP. Ontologies, as a part of the KMS represents domain knowledge in machine-readable form and provide a common language [3] for communication between various entities. It also offers modularity wherein a big domain can be broken down into several modules that can be reintegrated when needed. This paper focuses on the development of a domain ontology [15] for power systems. It also provides a framework for integrating this ontology with other domains like chemical engineering or logistics in an EIP. The developed ontology, OntoPowSys is evaluated using standard ontology metrics and its capabilities are demonstrated by implementing it in a smart system called the J-Park Simulator (JPS) [34].

The technical contributions of the paper are as follows:

- OntoPowSys: A domain ontology capable of formally representing Energy Management Strategies (EMS) in a wide industrial estate (like the JPS).
- The implementation of the ontology in the JPS which enables it to conduct various case studies by deploying computational agents.
- The application of the ontology for the automated generation of grid topology which is then utilised for power systems analysis like Optimal Power Flow (OPF).
- The application of the proposed ontology for studying cross domain interactions in a bio-diesel plant.

The rest of the paper is organised as follows. Section 2 describes the background and related work associated with the paper. Section 3 describes the methodology for ontology development and the evaluation metrics used for validating the ontology. Section 4 describes some of the major concepts defined in the OntoPowSys knowledge base. The following section describes the applications of the proposed ontology. Finally, section 6 provides the conclusion and future work.

2 Background and Related Work

2.1 Background

The concept of sharing and reusing technical knowledge using knowledge engineering began primarily from the KACTUS project [28]. The major objectives of the project were to create reusable knowledge sharing platforms and develop tools that could harness this information. The realisation of these objectives required a standardisation of knowledge in these domains. Ontologies which are "formal, explicit specification of a shared conceptualisation" were used to achieve these objectives [9]. Even though the scope of the KACTUS project was limited, their approach has been adopted by several domains like bio medical [1, 17], chemical engineering, electrical engineering [5].

An ontology comprises mainly of classes, properties, individuals, and axioms (ontology schema) [9]. Classes describe the categorisation of individuals and properties relate them. Axioms are used to express basic statements in the ontology by reusing classes and properties [10].

The methodology used for the development of OnotPowSys is adopted from [6, 29]. In this paper, the ontology is formally expressed using a Description Logics (DL) syntax. The resulting representation of ontology is termed as a knowledge base (KB) (see Section 6). The concept inclusion (CI) axioms in the KB define subsumption relations between concept names (classes in ontology). The role inclusion (RI) axioms define subsumption relation between role names (property chains in ontology). Finally, individual assertions include relations between individuals by using concept and role names. A set of CI and RI axioms is called a Tbox and a set of individual assertions can be called as an Abox [25].

2.2 Related Works

The primary use of ontologies in the power systems domain has been in the development of task or event ontologies for achieving specific objectives [23, 27]. The focus of the current paper, however is in the development of a domain ontology in power systems domain that is capable of studying multi domain interactions. The existing domain ontologies in the power systems can be broadly classified into two types. The first type focuses on specific areas in power systems domain ontologies in [2, 26]. These ontologies cannot be utilised for EMS at an estate level. Other types like [12] gives a detailed ontology for an electrical grid with a description of grid assets but are not suitable for multi-domain approaches (e.g. different classes of power generator, energy market aspects) [12].

To encapsulate the above approaches, an upper level ontology that can encompass all the broader concepts relevant to an EIP has been developed, called OntoEIP [34]. Such an ontology should be capable of interacting effectively with ontologies in other domains. There are several modules in electrical network domain of OntoEIP which should be expanded to work in tandem for better efficiency and management of resources.

2.2.1 OntoEIP

The OntoEIP ontology utilises the design philosophy of OntoCAPE: a tested, well defined ontology for the computer aided process engineering domain [20, 21]. The OntoCAPE is capable of representing knowledge in the chemical engineering domain. However, an ontology for EIP should implement additional modules from other domains like electrical engineering, logistics etc. OntoEIP uses abstraction layering to represent modules that are divided into different layers. Each layer contains just the information required for its function and all other information are passed to lower levels. The information is decomposed into subsystems in a bottom up approach where the lower levels have information of more specific nature. The OntoEIP ontology implements this abstraction and it is used in the KMS of the JPS.

2.2.2 J-Park Simulator (JPS)

The JPS is a multi-domain interactive simulation platform and expert system. It can serve as a common platform for establishing cross domain correlations in an EIP [34]. One of the major objectives of the JPS is to create a cyber-physical system wherein every entity in an EIP have their own virtual avatars. These virtual avatars will be capable of emulating the actual behaviour of the physical entity. They can be used to predict the behaviour of the EIP when there is a stimulus [32]. The JPS includes a KMS [33] which is capable of storing and analysing information from various domains like chemical, electrical, logistical. It also has a semantic web based agent framework [24] whose tasks can vary from querying of the knowledge graph to executing complex mathematical models [4].

3 Methodology

3.1 Ontology development

This section describes the methodology used in the development of OntoPowSys ontology. It uses OntoEIP upper layer modules as shown in Fig. 1 which illustrates the module hierarchy. Each block in Fig. 1 represents a module which is a collection of ontologies pertainingg to a similar notions. The blocks represented in the OntoEIP layer are the four technical systems defined in OntoEIP. OntoPowSys extends the classes and properties hierarchy defined in the electrical power system module of OntoEIP. The agent ontology extends the *Supporting Concepts* by creating a framework for executing mathematical

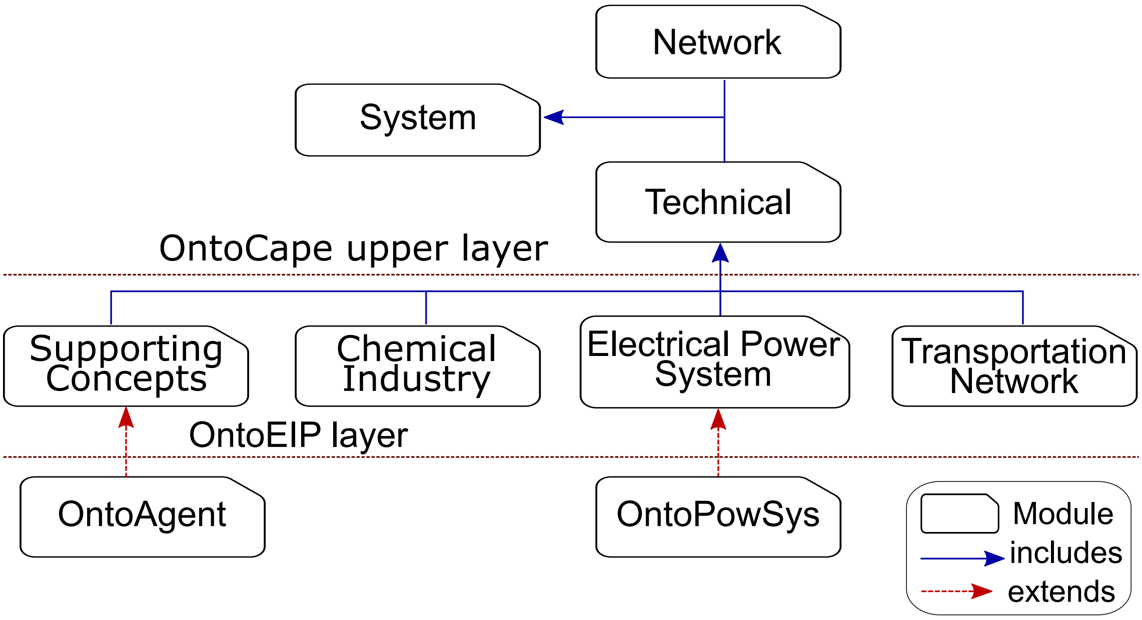


Figure 1: *Abstraction layering in the OntoPowSys ontology with the uppermost layer showing most generic concepts.*

models in the JPS.

OntoPowSys knowledge base (KB) borrows several axioms from OntoEIP. The KB is implemented in a machine-readable format using the Ontology Web Language (OWL2) [19]. Protégé [22], an ontology editor is used for the development and evaluation of the ontology. Once the ontology is developed, it is automatically populated using data from the JPS. The entire ontology is then evaluated for its quality and is then published on the JPS web portal.

3.2 Evaluation

Ontology development should include validation using suitable evaluation metrics as shown in Fig. 2. The evaluation process can be broadly classified into two steps. The first step is the evaluation of ontology schema and the second one is for ontology populated with data. The ontology schema was tested for its consistency (presence of any contradictions) using the HermiT reasoner [8] in Protégé and an overview of the process is given in Fig. 3. The schema was first tested locally and suitable modifications were made to make it consistent. This ontology is then published online. It was then tested for its coherency (measure of modularity) using a debugger plugin in Protege and was found to be coherent.

The ontology populated with the data was found to be consistent and coherent using the same procedure. The populated ontology was evaluated for additional ontology metrics as shown in Fig. 2 (step 5). It has to be noted that the generated data added to the OntoPowSys ontology do not utilise all the axioms defined in it. So, the evaluation metrics are applied to only those axioms.

The accuracy of the ontology was established by ensuring the inferred axioms in the

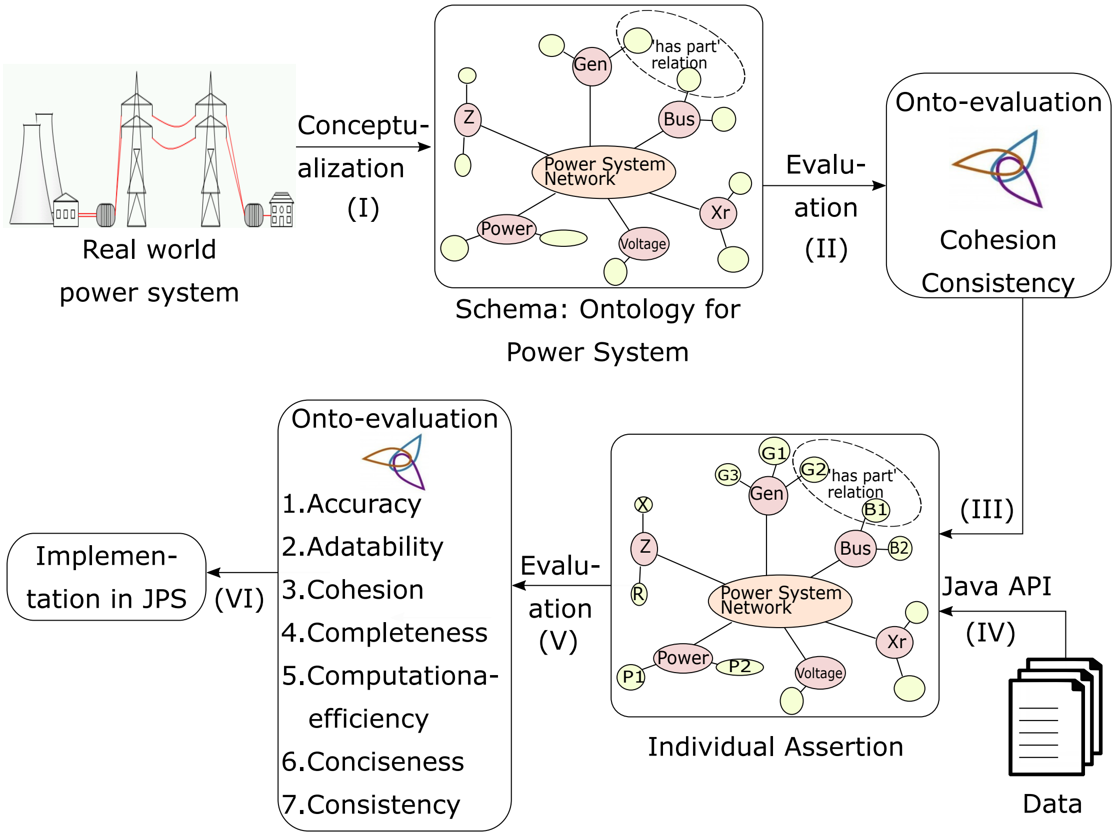


Figure 2: *OntoPowSys development process involves the conceptualisation of the power system domain into DL axioms which are implemented in Protége. The resulting ontology is then validated using suitable ontology metrics and published online.*

ontology matches with the domain expert's definitions. The accuracy of the inherited axioms from OntoEIP is not guaranteed. One such example is the *isComposedOf* role name defined in Table 11 of the Appendix. It was used to relate current with its angle and magnitude. As an example, take two different electrical lines with *Current*(c_1), *Current*(c_2), *CurrentAngle*(a_1), *CurrentMagnitude*(m_1), *and CurrentMagnitude*(m_2) as instances of their respective concept names, then:

$$\begin{aligned} &isComposedOf(c_1,a_1), isComposedOf(c_1,m_1),\\ &isComposedOf(c_2,a_1), isComposedOf(c_2,m_1), \end{aligned} \implies sameAs(c_1,c_2) \tag{1}$$

$$\begin{aligned} &isComposedOf(c_1,a_1), isComposedOf(c_1,m_1),\\ &isComposedOf(c_2,a_2), isComposedOf(c_2,m_2), \end{aligned} \implies \neg(sameAs(c_1,c_2)) \tag{2}$$

Condition 1 can be easily satisfied by making the role name *isComposedOf* inverse functional, i.e. $\top \sqsubseteq \leqslant 1 isComposedOf^{-}.\top$. But this will lead to violation of condition 2 since the inverse functionality of the role name will assume $sameAs(m_1,m_2)$ and will identify *sameAs*(c_1, c_2) as well. This problem was rectified by adding two role names *hasAngle* and *hasMagnitude* subsumed by *isComposedOf* in the OntoPowSys KB.

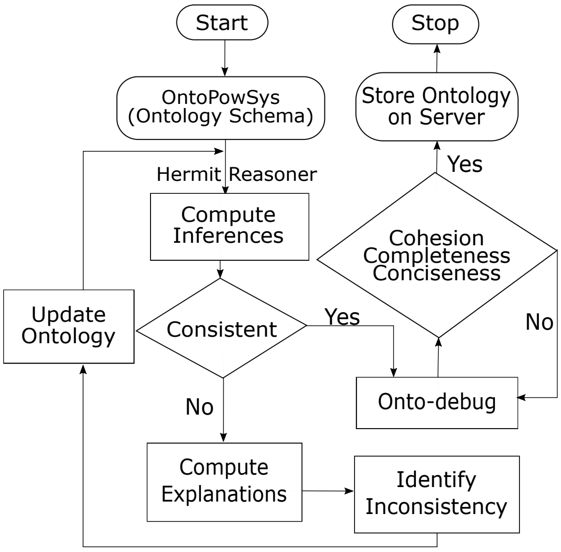


Figure 3: *Evaluation procedure for ontology schema.*

Another metric is adaptability which refers to the ability of the ontology to adapt based on the application [11]. OntoPowSys ontology was developed to incorporate the cross-domain interactions in an EIP and is capable of tending to a wide range of aspects in an EIP as will be evident from the use case scenarios demonstrated in section 5. One such example is the space and time ontology which is imported in OntoPowSys. It contains classes and properties relevant to the location and time definitions of a system and could be used for comparing instances at different points of time like in the case of optimal dispatch of resources.

The conciseness of the ontology refers to its ability to represent the domain without any redundant axioms and minimum ontological commitment [11]. OntoPowSys being a combination of several ontologies contain axioms that are not entirely relevant to a power system. One such example is the SI_units ontology which contains the units of measurements for most of the physical quantities. Although OntoPowSys requires units of measurement related to power or current or voltage, units like mass or force are irrelevant. On top of that, some classes defined in the OntoPowSys are redundant at present but might become relevant in the future. One such example is the *EnergyStorageSystem* class (see Table 8) which does not have any instances at present due to lack of data in the JPS.

Computational efficiency refers to the expressivity of the ontology in DL and the ability of the reasoners to classify the ontology. Protégé classified the DL expressivity of the OntoPowSys as SHOIQ which means that HermiT reasoner can successfully classify the ontology.

4 OntoPowSys: A domain ontology for power systems

This section gives a brief description of the important concepts and roles defined in the OntoPowSys KB. Table 4 of the Appendix provides a complete list of the terms defined in OntoPowSys ontology. In the KB, *PowerSystem* concept name is subsumed by *CompositeSystem* concept name in OntoEIP (see axiom 1 of Table 6).

A typical power system consists of a set of power generators that are connected to their respective power loads through an electrical network consisting of buses. It also has devices like power converters to manage the quality and quantity of power flow. We are modelling them using concept inclusion axioms such as axiom 1 of Table 8.

These subsystems follow the conventional structure of generation, transmission and consumption. The generation is expressed as *PowerGenerator* concept name (axioms 9 and 10 of Table 6) that is subsumed by two other concepts *DispatchableUnits* and $Non-DispatchableUnits$, based on the nature of their generation. These generators are considered as power sources with a set of rated electrical parameters. This is expressed by using domain and range restrictions on *hasProperty* role name, as well as RI axioms and restricting it as inverse functional (see axioms *hasProperty* role name in Table 11).

Similarly, power loads are considered as generic power sinks with a set of rated parameters. OntoPowSys is used to represent in detail all the possible types of electrical consumption from individual devices level (e.g. pumps, motors etc.) to clusters of loads (e.g. districts, buildings etc.). Also, some concepts allow identification of critical loads that have to be supplied at any time as well as controllable loads (axioms 3 and 4 of *PowerLoad* concept name in Table 8). The controllable loads offer some flexibility in power dispatch problem in the framework of demand response. Such frameworks can be considered as a promising way to increase the integration of renewable energy sources and reduce the use of dirtier or more expensive unit to supply the peak load.

Bus nodes constitute the transmission network of a power system and are required to develop the electrical network topology. Power injection and consumption at a bus helps to compute the power flow. Considering a bus node as a physical point of connection between two electrical devices allow for a more generic and easy problem formulation when dealing with flow analysis. The concept name *ElectricalLine* (axioms 7 and 8 of Table 8) connects the buses using *hasInput* and *hasOutput* role names that are subsumed by *isConnectedTo* role name (see axioms of *isConnectedTo* role name in Table 11).

Energy storage which is a subsystem of power system allows more flexibility in order to modify a load curve (e.g. peak shaving, load shifting) or enhance the integration of renewables (e.g. solar generation smoothing). Based on concept definition of the state of charge (see axioms 10 to 12 of *EnergyStorageSystem* concept name in Table 8), these types of equipment can behave either as power sources or power sinks within their operating limits. Several kinds of technologies like electrochemical storage (e.g. battery) or a mechanical one (e.g. compressed air) are available for energy storage.

Power converters realise the energy flow from one form of electrical energy to another. Transformers allow conversion from one AC voltage level to another while typical converters realise DC and AC conversion. The number of properties attached to those converters is higher than the other grid components as the ratings refer both the input and output side (see all CIs in Table 8). Additionally, one important property of the converters is whether they allow bi-directional flow of energy like batteries or uni-directional flow like low voltage supply transformers (see axiom 5 of *PowerConverter* concept name in Table 8).

The capital and operating costs related to all these *ElectricalEquipment* are modelled using the *hasCost* role name and role names subsumed by it. These role names are restricted to be inverse functional (see CI axiom 2 of *hasCost* role name in Table 9) in nature to avoid multiple concepts from having the same costs. The axioms defined in the KB does not go deep down into the electrical engineering domain with the description of their topology and regulation loops which should be part of dedicated knowledge bases.

Tables 8 and 9 attached in the appendix gives complete information about the CI and RI axioms defined in OntoPowSys KB. Table 11 in the appendix section of the paper describes various OntoEIP CI and RI axioms imported by OntoPowSys.

Once the OntoPowSys KB is made alive using the OWL2 ontology language it is populated by generating individual assertions using the data available from JPS. The application of the ontology to carry out various studies is described in the following section.

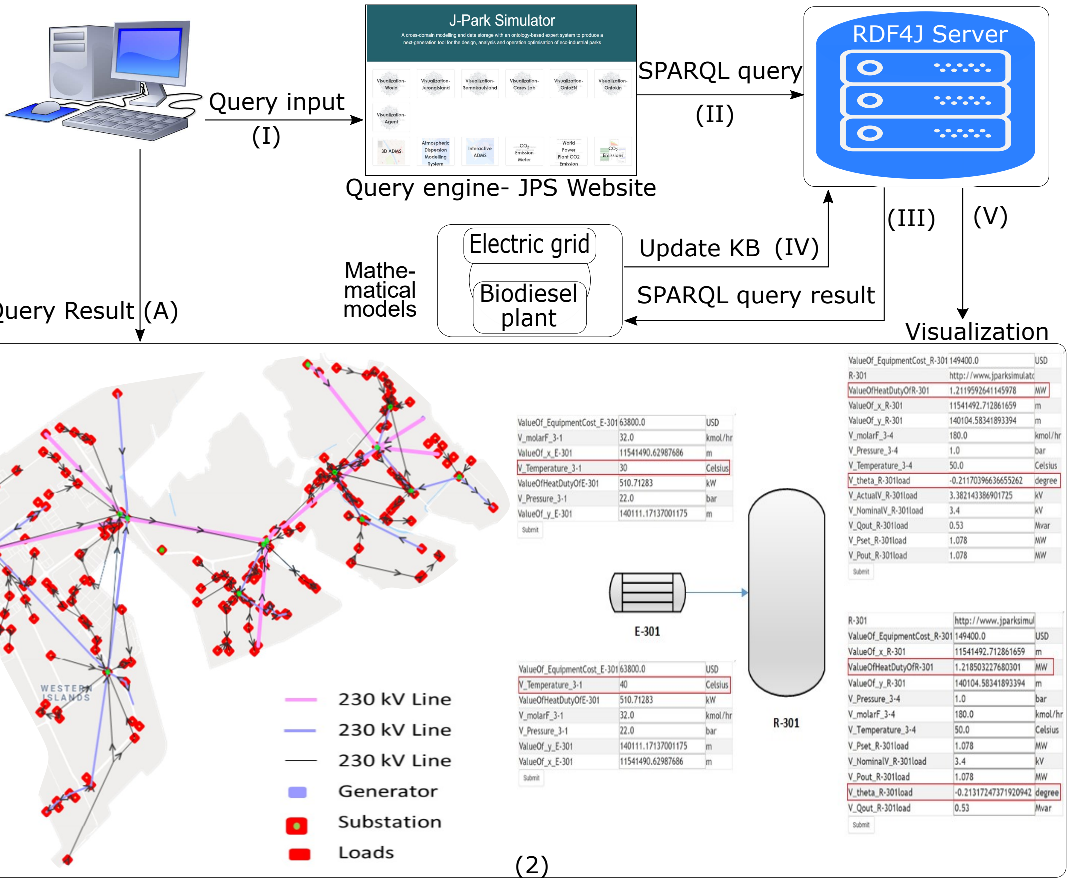
5 Application of OntoPowSys in JPS

The OntoPowSys ontology has then been integrated into the JPS, wherein the knowledge is utilised to carry out various studies and simulations. Fig. 4 describes the architecture for conducting case studies in JPS which has a set of web services available at `http://www.theworldavatar.com/`. These web services offer a wide variety of agents that can execute specific tasks. The user can input the relevant information depending on the agent type. Based on these inputs, a series of SPARQL queries are executed on the JPS knowledge graph. These queries collect the necessary and sufficient information required for the execution of the mathematical models. These data are fed to the respective models and they are executed. The results from the model update the knowledge graph and are visualised using suitable tools. The following subsections give specific examples of the agents available in the JPS. The first agent computes the problem of Optimal Power Flow (OPF) in the JPS grid while the second one is used to study the cross-domain interactions in a biodiesel plant.

5.1 Optimal Power Flow (OPF) agent

The objective is to optimise the load supply with the minimum generation cost while fulfilling grid constraints i.e. bus voltages and line current limit. In such problems, the power produced by the generators are the only degrees of freedom and convergence is not ensured if the system is not designed appropriately.

Solving an OPF problem requires a well-defined grid topology. The topology provides information about the connections between the bus nodes and different equipment like power generators, power loads etc. Usually, they are created manually and needs to be



ValueOf_EquipmentCost_E-301	63800.0	USD
V_molarF_3-1	32.0	kmol/hr
ValueOf_x_E-301	11541490.62987686	m
V_Temperature_3-1	30	Celsius
ValueOfHeatDutyOfE-301	510.71283	kW
V_Pressure_3-1	22.0	bar
ValueOf_y_E-301	140111.17137001175	m

Submit

ValueOf_EquipmentCost_E-301	63800.0	USD
V_Temperature_3-1	40	Celsius
ValueOfHeatDutyOfE-301	510.71283	kW
V_molarF_3-1	32.0	kmol/hr
V_Pressure_3-1	22.0	bar
ValueOf_y_E-301	140111.17137001175	m
ValueOf_x_E-301	11541490.62987686	m

Submit

ValueOf_EquipmentCost_R-301	149400.0	USD
R-301	http://www.jparksimulato	
ValueOfHeatDutyOfR-301	1.2119592641145978	MW
ValueOf_x_R-301	11541492.712861659	m
ValueOf_y_R-301	140104.58341893394	m
V_molarF_3-4	180.0	kmol/hr
V_Pressure_3-4	1.0	bar
V_Temperature_3-4	50.0	Celsius
V_theta_R-301load	-0.21170396636655262	degree
V_ActualV_R-301load	3.382143386901725	kV
V_NominalV_R-301load	3.4	kV
V_Qout_R-301load	0.53	Mvar
V_Pset_R-301load	1.078	MW
V_Pout_R-301load	1.078	MW

Submit

R-301	http://www.jparksimul	
ValueOf_EquipmentCost_R-301	149400.0	USD
ValueOf_x_R-301	11541492.712861659	m
ValueOfHeatDutyOfR-301	1.218503227680301	MW
ValueOf_y_R-301	140104.58341893394	m
V_Pressure_3-4	1.0	bar
V_molarF_3-4	180.0	kmol/hr
V_Temperature_3-4	50.0	Celsius
V_Pset_R-301load	1.078	MW
V_NominalV_R-301load	3.4	kV
V_Pout_R-301load	1.078	MW
V_theta_R-301load	-0.21317247371920942	degree
V_Qout_R-301load	0.53	Mvar

Submit

Figure 4: *Implementation of OntoPowSys in JPS. The user input on the website triggers the respective agents which are executed using the data from SPARQL queries on the knowledge graph. The outputs from the agents are updated on the knowledge graph and they are visualised.*

Table 1: *Abox representing grid topology for OPF.*

$UndergroundCable(ELine-115)$
$OutputVoltage(OutputVoltage\_ELine-115)$
$hasVoltageOutput(ELine-115, OutputVoltage\_ELine-115)$
$VoltageMagnitude(OutputVoltageMagnitude\_ELine-115)$
$VoltageAngle(OutputVoltageAngle\_ELine-115)$
$isComposedOf(OutputVoltage\_ELine-115, OutputVoltageMagnitude\_ELine-115)$
$isComposedOf(OutputVoltage\_ELine-115, OutputVoltageAngle\_ELine-115)$
$BusNode(EBus-085)$,
$BusNode(EBus-097)$,
$BusNode(EBus-001)$
$hasOutput(ELine-115, EBus-085)$
$hasInput(ELine-115, EBus-097)$
$Cylinder(Shape\_ELine-115)$
$EdgeLength(Length\_ELine-115)$
$hasLength(Shape\_ELine-115, Length\_ELine-115)$
$hasValue(Length\_ELine-115, V\_Length\_ELine-115)$
$numericalValue(V\_Length\_ELine-115, 1.1436772)$
$hasUnitOfMeasure(V\_Length\_ELine-115, km)$
$PrefixedDerivedUnit(km)$
$PowerGenerator(EGen-001)$
$hasOutput(EGen-001, EBus-001)$

modified every time there is a change in the grid. In this application, OntoPowSys is used to represent the grid topology. Table 1 represents a section of the ABox assertions that depicts the topology. It describes how an individual such as $ELine-115$ (axiom 1 of Table 1) is related to other equipment in the power system. Axioms 2 to 7 in the table represents how the electrical parameters of the individual are defined. It needs to be noted that only selected section of ABox individual assertions is listed in the Table. The role names *hasInput* and *hasOutput* connects $ELine-115$ with its respective inputs and outputs (axioms 9 and 10 of Table 1). The remaining axioms describe the physical properties of the electrical line that are required to derive the topology.

The current electrical network in JPS consists of 208 buses and 216 branches. The OPF problem is executed using the "Power Flow Agent" in the JPS. It uses Python library PyPower, a library for solving the OPF problem. Conventional numerical technique such as Newton-Raphson method is being utilised by the PyPower for computation. The role of the "Power Flow Agent" can be divided into three main actions:

1. Generate the OPF inputs after querying the JPS knowledge graph in accordance with the library format that consists of matrices for bus, line and generators. For each set of parameters (e.g. bus rated voltage), input variable (e.g. bus active/reactive demands) and state variables (e.g. bus voltage) are defined. The power consumed or injected by bus are derived from the load connected to them.

2. Run the OPF model using the python module.

3. Read the outputs and update the JPS knowledge graph with the computed state variables for the line current/power losses, bus voltages and generated powers. It should be noted that some variables such as bus voltage magnitude and angles are both input variables (required by the PyPower format) and updated value (output from the OPF).

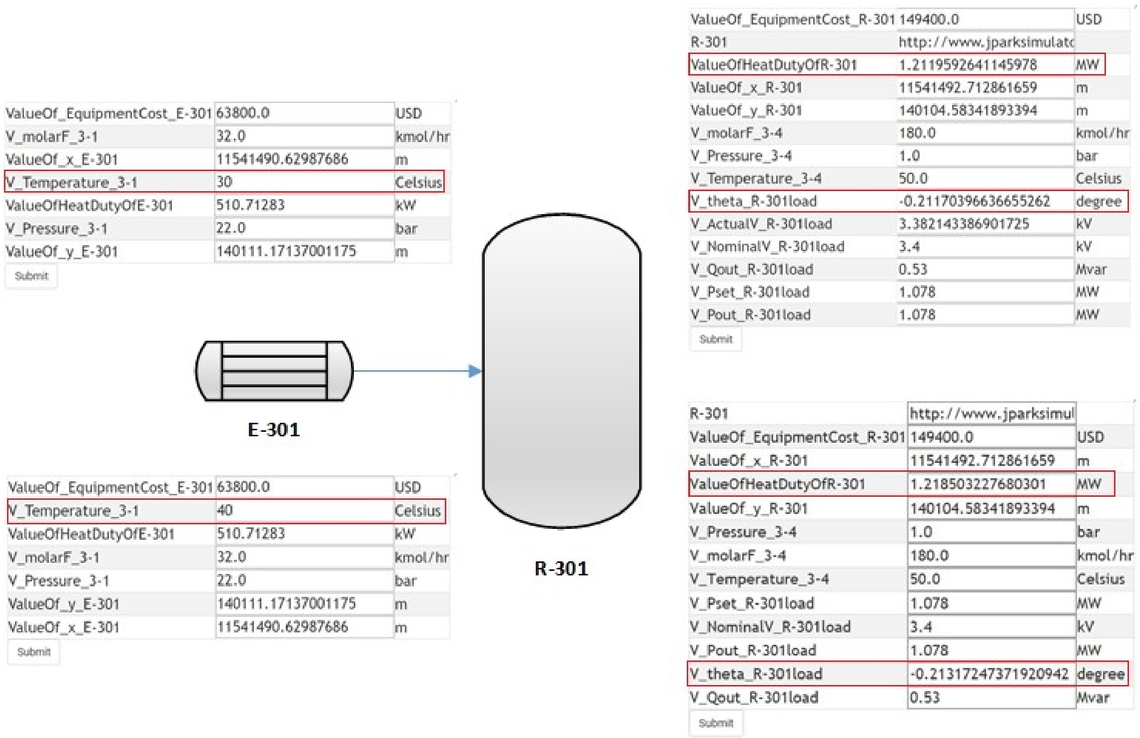
Once the power flow agent finishes execution of the model and updates the results in the JPS knowledge graph, the visualisation of the power grid is semiautomatically completed. The geographical information pertaining to the electrical equipment implemented in the *spaceandtime* ontology of OntoEIP is used for this task. A sample of the final visualisation is shown in part (1) of Fig. 4

5.2 Bio diesel plant simulation

This use case showcases the OntoPowSys's ability for cross domain interactions. It takes the example of a biodiesel plant in JPS (see Fig. 4). The chemical and electrical domain knowledge of the plant is represented using respective modules of the OntoEIP part of the JPS knowledge graph. The interactions in the chemical domain of the system are modelled using an Aspen plus model and the electrical interactions are modelled using a PowerWorld. These interactions are not independent of each other as a change in the chemical engineering domain can cause a change in the electrical engineering domain and vice versa.

Table 2: *Abox representing the inputs for the case studies.*

Case	Inputs
Case 1	$Temperature.(Temperature\_3\_1)$
	$ScalarValue(V\_Temperature\_3\_1)$
	$hasValue(Temperature\_3\_1, V\_Temperature\_3\_1)$
	$numericalValue(V\_Temperaure\_3\_1, 30)$
	$hasUnitOfMeasure(V\_Temperaure\_3\_1, Celsius)$
	$SI\_DerivedUnit(Celsius)$
Case 2	$numericalValue(V\_Temperaure\_3\_1, 40)$



ValueOf_EquipmentCost_E-301	63800.0	USD
V_molarF_3-1	32.0	kmol/hr
ValueOf_x_E-301	11541490.62987686	m
V_Temperature_3-1	30	Celsius
ValueOfHeatDutyOfE-301	510.71283	kW
V_Pressure_3-1	22.0	bar
ValueOf_y_E-301	140111.17137001175	m

ValueOf_EquipmentCost_E-301	63800.0	USD
V_Temperature_3-1	40	Celsius
ValueOfHeatDutyOfE-301	510.71283	kW
V_molarF_3-1	32.0	kmol/hr
V_Pressure_3-1	22.0	bar
ValueOf_y_E-301	140111.17137001175	m
ValueOf_x_E-301	11541490.62987686	m

ValueOf_EquipmentCost_R-301	149400.0	USD
R-301	http://www.jparksimulatc	
ValueOfHeatDutyOfR-301	1.2119592641145978	MW
ValueOf_x_R-301	11541492.712861659	m
ValueOf_y_R-301	140104.58341893394	m
V_molarF_3-4	180.0	kmol/hr
V_Pressure_3-4	1.0	bar
V_Temperature_3-4	50.0	Celsius
V_theta_R-301load	-0.21170396636655262	degree
V_ActualV_R-301load	3.382143386901725	kV
V_NominalV_R-301load	3.4	kV
V_Qout_R-301load	0.53	Mvar
V_Pset_R-301load	1.078	MW
V_Pout_R-301load	1.078	MW

R-301	http://www.jparksimul	
ValueOf_EquipmentCost_R-301	149400.0	USD
ValueOf_x_R-301	11541492.712861659	m
ValueOfHeatDutyOfR-301	1.218503227680301	MW
ValueOf_y_R-301	140104.58341893394	m
V_Pressure_3-4	1.0	bar
V_molarF_3-4	180.0	kmol/hr
V_Temperature_3-4	50.0	Celsius
V_Pset_R-301load	1.078	MW
V_NominalV_R-301load	3.4	kV
V_Pout_R-301load	1.078	MW
V_theta_R-301load	-0.21317247371920942	degree
V_Qout_R-301load	0.53	Mvar

Figure 5: *Bio-diesel plant simulation results in JPS. The tables on the left side of the reactor represent the inputs to the simulations and the tables on the right shows the results.*

The biodiesel plant has a reactor (R-301) which is implemented in OntoEIP using the CI axioms defined of *StirredTank* concept name given in Table 10 of Appendix. The reactor is equipped with an electric heater in order to maintain the required outlet temperature (refer to CI axioms of *Temeprature* concept name given in Table 10 of Appendix). The user can input the desired value of the reactor outlet temperature on the JPS web portal. This value of the reactor temperature is fed into the Aspen model which determines the heat duty required to increase the current temperature to the required value. The heat duty is implemented using the CI axioms of *Masstransfercoefficent* concept name given in Table 10 of Appendix. This energy requirement is then fed into the PowerWorld model which determine the electrical requirements for the heater. The electrical load requirements are implemented using the first and third CI axioms of *PowerLoad* concept name given in Table 8. These models are run in the background and the results are displayed on the JPS web portal (see Fig. 4, part (2)).

Table 3: *Abox representing the solutions from the case studies.*

Case	Outputs
Case 1	$MassTransferCoefficient(HeatDutyOfR-301)$
	$ScalarValue(ValueOfHeatDutyOfR-301)$
	$hasValue(HeatDutyOfR-301, ValueOfHeatDutyOfR-301)$
	$numericalValue(ValueOfHeatDutyOf-R-301, 1.211)$
	$hasUnitOfMeasure(ValueOfHeatDutyOfR-301, MW)$
	$SI\_DerivedUnit(MW)$
	$PowerLoad(R-301load)$
	$Voltage(Voltage\_R-301load)$
	$VoltageAngle(theta\_R-301load)$
	$ScalarValue(V\_theta\_R-301load)$
	$hasActualVoltage(R-301load, Voltage\_R-301load)$
	$isComposedOf(Voltage\_R-301load, theta\_R-301load)$
	$hasValue(theta\_R-301load, V\_theta\_R-301load)$
	$numericalValue(V\_theta\_R-301load, -.2117)$
	$hasUnitOfMeasure(V\_theta\_R-301load, degree)$
	$SI\_DerivedUnit(degree)$
Case 2	$numericalValue(ValueOfHeatDutyOfR-301, 1.218)$
	$numericalValue(V\_theta\_R-301load, -.2113)$

Once the user inputs the reactor temperature on the JPS web portal, the value is updated in the OntoEIP part of the JPS knowledge graph. Table 2 represents the set of Abox assertions thus created. The case study is presented as two cases with different values for reactor outlet temperatures. The first two individual assertions in Table 2 define the temperature of the reactor outlet stream (3_1). The next three assertions establish relations between the reactor ID and its temperatures. For case 2, only one assertopn that defines the numerical value of temperature is defined as other assertions will be the same. Table 3 represents the set of Abox assertions that are generated once the simulation is executed. The first six assertions of Table 3 defines the heat duty of the reactor and the rest defines the electrical load requirements of the reactor. Only a section of the results is presented in the tables. The rest of the results can be seen in Fig 5.

6 Conclusion and Future work

This paper illustrates the development and applications of OntoPowSys, a domain ontology for power systems in an EIP. It was developed to be a part of JPS, a web based software platform for integrating data, knowledge and models in an EIP. JPS includes several agents that operate using the knowledge from the ontologies. Two of such agents and their architecture for execution has been described in the paper. The case studies conducted using these web agents prove that the proposed approach increases inter operability between various domains. This operability is supported by HermiT reasoner that keeps the JPS knowledge graph consistent. It also provides a higher degree of automation and allows data distribution over the web.

At its full potential the proposed system will be capable realising the concept of smart factories that can operate with minimum human interference. This would however require much more expressive ontologies combined with other Industry 4.0 technologies.

Our future work in this area can be divided into two parts. The first part will focus on improving expressivity of the ontology by adding more contextual details and employing more stringent validation metrics. This includes automating the process of ontology development and generating specialised ontologies for electrical engineering domains. The second part will focus on developing rule-based systems on the ontology that are capable of decision making without the use of any external agents.

Acknowledgements

This project is funded by the National Research Foundation(NRF), Prime Ministers Office, Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) programme. Markus Kraft gratefully acknowledges the support of the Alexander von Humboldt foundation.

List of abbreviations

EIP	**E**co **I**ndustrial **P**ark
JPS	**J-P**ark **S**imulator
OWL	**O**ntology **W**eb **L**anguage
CI	**C**oncept **I**nclusion
RI	**R**ole **I**nclusion
OPF	**O**ptimal **P**ower **F**low
KB	**K**nowledge **B**ase
EMS	**E**nergy **M**anagement **S**trategies
KMS	**K**nowledge **M**anagement **S**ystems

Appendix

A. Term definitions

Table 4: *OntoPowSys term definitions.*

Term name	**Definition**
Power generator	A generator is a rotating device that converts kinetic energy into electrical energy for use in an external circuit.
Power load	A power load is an electrical component or portion of a circuit that consumes electric power.
Energy storage system	Energy storage is the capture of energy produced at one time for use at a later time.
Transmission network	It is an interconnected network for delivering electricity from generating stations to loads.
Power converter	Power converter is an electronic switching device for converting electrical energy from one form to another such as AC-DC, DC-AC or change in frequency.
Power transformer	The Power transformer is a static device,that transfers electrical energy from one circuit to another. It is commonly used to step up or step down voltage levels.
Rectifier	A static electrical device which converts an alternating current into a direct current by allowing a current to flow through it in one direction only.
Inverter	A static electrical device which converts direct current into alternating current.
Dispatchable units	Dispatchable units of generation refers to sources of electricity that can be used on demand and dispatched at the request of power grid operators, according to market needs.
Non dispatchable units	A non-dispatchable source of electricity generate electrical energy but cannot be turned on or off in order to meet societies fluctuating electricity needs.
Energy source	A source from which useful energy can be extracted or recovered either directly or by means of a conversion or transformation process.
Current type	Electric current comes in two varieties: alternating current and direct current, abbreviated as AC and DC.

Voltage Regulator	A voltage regulator is a system designed to automatically maintain a constant voltage level.
Bus node	A bus network is an arrangement in a local area network (LAN) in which each node (workstation or other device) is connected to a main cable or link called the bus.
Electrical Line	An electrical line is a cable (above ground or underground), along which electricity is passed to an area or building.
Switch	A switch is an electrical component that can "make" or "break" an electrical circuit, interrupting the current or diverting it from one conductor to another.
Energy meter	An energy meter is a device that measures the amount of electric energy consumed by a residence, a business, or an electrically powered device.
Rated power	The rated power of an equipment is the highest power input allowed to flow through particular equipment.
Rated efficiency	Rated energy efficiency is a ratio of the amount of energy that is usefully used compared with the total energy input to obtain that useful output.
Rated current	Rated current is that current for which the equipment or the machine has been designed.
Rated voltage	Rated voltage is that voltage for which the equipment or the machine has been designed.
Active power	The power which is actually consumed or utilized in an AC Circuit is called True power or Active Power. It is measured in terms of watt.
Reactive power	Reactive power is the part of complex power (Apparent Power) that corresponds to storage and retrieval of energy rather than consumption of power.It results from dissipation from inductive and capacitive loads, measured in terms of VAR (volt-ampere reactive).

Table 5: *OntoAgent terms definitions.*

Term name	**Definition**
Agent	An autonomous entity which acts, directing its activity towards achieving goals, upon an environment using observation through sensors and consequent actuators.

Mathematical model	A mathematical model is a description of a system using mathematical concepts and language.
Visualization	The representation of an object, situation, or set of information as a chart or other image.
I/O Agent	Reads or write information from the model to the knowledge base.
Output variable	Represents input to a model.
Input variable	Represents the outputs from the model.
Parameter	An optimization parameter (or a decision variable, in the terms of optimization) is a model parameter to be optimized.

B. Concept defintions

Table 6: *OntoPowSys concept hierarchy.*

Term name	**Concepr definitions**
Power system	$PowerSystem \sqsubseteq CompositeSystem$
Electrical equipment	$ElectricalEquipment \sqsubseteq CompositeSystem$
Bus node	$BusNode \sqsubseteq ElectricalEquipment$
Electricity line	$ElectricityLine \sqsubseteq System$
Energy meter	$EnegryMeter \sqsubseteq ElectricalEquipment$
Power load	$PowerLoad \sqsubseteq ElectricalEquipment$
Energy storage system	$EnergyStorageSystem \sqsubset ElectricalEquipment$
Power generator	$PowerGenerator \sqsubseteq ElectricalEquipment$
Dispatchable units	$DispatchableUnits \sqsubseteq PowerGenerator$
Non-dispatchable units	$NonDispatchableUnits \sqsubseteq PowerGenerator$
Power converter	$PowerConverter \sqsubseteq ElectricalEquipment$
Inverter	$Inverter \sqsubseteq PowerConverter$
Power transformer	$PowerTransformer \sqsubseteq PowerConverter$
Rectifier	$Rectifier \sqsubseteq PowerConverter$
Power load	$PowerLoad \sqsubseteq ElectricalEquipment$

Building	$Building \sqsubseteq PowerLoad$
District	$District \sqsubseteq PowerLoad$
Electrical pump	$ElectricalPump \sqsubseteq PowerLoad$
Electronics	$Building \sqsubseteq Electronics$
Power plant	$PowerPlant \sqsubseteq ElectricalEquipment$
Co-generation plant	$CogenerationPlant \sqsubseteq Powerplant$
Fossil fuel plant	$FossilFuelPlant \sqsubseteq Powerplant$
Hydroelectric plant	$HydroElectricPlant \sqsubseteq Powerplant$
Nuclear plant	$NuclearPlant \sqsubseteq Powerplant$
Renewable plant	$RenewablePlant \sqsubseteq Powerplant$
Sub station	$SubStation \sqsubseteq ElectricalEquipment$
Distribution sub station	$DistributionSubstation \sqsubseteq Substation$
Transmission sub station	$TransmissionSubstation \sqsubseteq Substation$
Switch	$Switch \sqsubseteq ElectricalEquipment$
Voltage regulator	$VoltageRegulator \sqsubseteq ElectricalEquipment$
Overhead line	$OverheadLine \sqsubseteq ElectricalLine$
Underground cable	$UndergroundCable \sqsubseteq ElectricalLine$
Battery	$Battery \sqsubseteq ElectricalLine$
Lead acid battery	$LeadAcidBattery \sqsubseteq Battery$
Lithium ion battery	$LithiumIonBattery \sqsubseteq Battery$
Nickel Cadmium battery	$NickelCadmiumBattery \sqsubseteq Battery$
Compressed air	$CompressedAir \sqsubseteq ElectricalLine$
Fly wheel	$FlyWheel \sqsubseteq ElectricalLine$
Hydrogen storage	$HydrogenStorage \sqsubseteq ElectricalLine$
Hydro pump	$PumpHydro \sqsubseteq ElectricalLine$
Photovoltaic panel	$PhotovoltaicPanel \sqsubseteq ElectricalLine$
Voltage	$Voltage \sqsubseteq PhysicalQuantity$
Current	$Current \sqsubseteq PhysicalQuantity$
Current type	$Voltage \sqsubseteq ConstantProperty$
Rated Current	$RatedCurrent \sqsubseteq ConstantProperty$

Rated power	$RatedPower \sqsubseteq ConstantProperty$
Rated frequency	$RatedFrequency \sqsubseteq ConstantProperty$
Rated voltage	$RatedVoltage \sqsubseteq ConstantProperty$
Rated power factor	$RatedPowerFactor \sqsubseteq ConstantProperty$
Frequency	$Frequency \sqsubseteq ScalarQuantity$
Power factor	$PowerFactor \sqsubseteq ScalarQuantity$
Reactive power	$Reactivepower \sqsubseteq ScalarQuantity$
Active power	$ActivePower \sqsubseteq ScalarQuantity$
Energy price	$EnergyPrice \sqsubseteq ScalarQuantity$
State of charge	$StateOfCharge \sqsubseteq ScalarQuantity$
Voltage angle	$VoltageAngle \sqsubseteq ScalarQuantity$
Voltage magnitude	$VoltageMagnitude \sqsubseteq ScalarQuantity$
Electrical network costs	$ElectricalNetworkCosts \sqsubseteq ScalarQuantity$
Accumulated costs	$AccumulatedCosts \sqsubseteq ScalarQuantity$
Capital expenditure costs	$CapitalExpenditureCosts \sqsubseteq ScalarQuantity$
Individual costs	$IndividualCosts \sqsubseteq ScalarQuantity$
Operational expenditure costs	$OperationalExpenditureCosts \sqsubseteq ScalarQuantity$
Shutdown costs	$Shutdowncosts \sqsubseteq OperationalExpenditureCosts$
Startup costs	$StartupCosts \sqsubseteq OperationalExpenditureCosts$
Power amount	$PowerAmount \sqsubseteq Device$
System component process	$SystemComponentProcess \sqsubseteq Device$
Power consumption	$PowerConsumption \sqsubseteq SystemComponentProcess$
Power equipment connection	$PowerEquipment \sqsubseteq SystemComponentProcess$
Power flow	$EnergyMetering \sqsubseteq SystemComponentProcess$
Power generation	$PowerGeneration \sqsubseteq SystemComponentProcess$
Voltage regulation	$VoltageRegulation \sqsubseteq SystemComponentProcess$

Table 7: *OntoAgent concept hierarchy.*

Term name	**Concept defintions**

Branch parameter	$BranchParameter \sqsubseteq Parameter$
Bus node parameter	$BusNodeParameter \sqsubseteq Parameter$
Generator parameter	$GeneratorParameter \sqsubseteq Parameter$
Branch state variable	$BranchParameter \sqsubseteq StateVariable$
Bus node state variable	$BusStateVariable \sqsubseteq StateVariable$
Generator state variable	$GeneratorStateVariable \sqsubseteq StateVariable$

C. OntoPowSys Axioms

Table 8: *CI axioms in OntoPowSys.*

Term name	**Axioms**
Power system	$PowerSystem \sqsubseteq \exists hasSubsystem.(BusNode \sqcup ElectricalLine \sqcup EnergyStorageSystem \sqcup PowerConverter \sqcup PowerGenerator \sqcup PoweLoad \sqcup Switch \sqcup VoltageRegulator)$
Electrical equipment	$ElectricalEquipment \sqsubseteq \exists hasCost.CapitalExpenditureCost$ $ElectricalEquipment \sqsubseteq \exists\ hasCost.OpearationalExpenditureCost$
Bus node	$BusNode \sqsubseteq \forall hasActualVoltage.ActualVoltage$ $BusNode \sqsubseteq \forall hasCurrentType.CurrentType$ $BusNode \sqsubseteq \forall hasFrequency.Frequency$ $BusNode \sqsubseteq \forall ElectricalLine \sqcap hasPowerInjection.(ActivePowerInjection \sqcup ReactivePowerInjection)$ $BusNode \sqsubseteq \forall hasRatedFrequency.RatedFrequency$ $BusNode \sqsubseteq \forall hasRatedVoltage.RatedVoltage$ $BusNode \sqsubseteq \forall realizes.PowerEquipmentConnection$

Electrical line	$ElectricalLine \sqsubseteq \forall hasActivePowerInput.InputActivePower$ $ElectricalLine \sqsubseteq \forall hasActivePowerLoss.ActivePowerLoss$ $ElectricalLine \sqsubseteq \forall hasActivePowerOutput.OutputActivePower$ $ElectricalLine \sqsubseteq \forall hasCurrent.Current$ $ElectricalLine \sqsubseteq \forall hasCurrentType.CurrentType$ $ElectricalLine \sqsubseteq \forall hasFrequency.Frequency$ $ElectricalLine \sqsubseteq \forall hasInput.BusNode$ $ElectricalLine \sqsubseteq \forall hasOutput.BusNode$
Overhead line	$OverheadLine \sqsubseteq \forall hasLineGeometry.LineGeometry$
Underground cable	$UndergroundCable \sqsubseteq \forall hasActivePowerInput.InputActivePower$
Energy meter	$EnergyMeter \sqsubseteq \forall hasCurrentType.CurrentType$ $EnergyMeter \sqsubseteq \forall hasRatedCurrent.RatedCurrent$ $EnergyMeter \sqsubseteq \forall hasRatedFrequency.RatedFrequency$ $EnergyMeter \sqsubseteq \forall hasRatedVoltage.RatedVoltage$ $EnergyMeter \sqsubseteq \forall hasSamplingTime.SamplingTime$ $EnergyMeter \sqsubseteq \forall isConnectedTo(BusNode \sqcup ElectricalLine \sqcup EnergyStorageSystem \sqcup PowerGenerator \sqcup PowerLoad \sqcup VoltageRegulator)$ $EnergyMeter \sqsubseteq \forall observes.(CumulatedEnergy \sqcap EnergyPrice)$ $EnergyMeter \sqsubseteq \forall realizes.EnergyMetering$

Energy storage system	$EnergyStorageSystem \sqsubseteq \forall hasActivePowerGenerated.GeneratedActivePower$ $EnergyStorageSystem \sqsubseteq \forall hasActualVoltage.Voltage$ $EnergyStorageSystem \sqsubseteq \forall hasCurrent.Current$ $EnergyStorageSystem \sqsubseteq \forall hasCurrentType.CurrentType$ $EnergyStorageSystem \sqsubseteq \forall hasEfficiency.Efficiency$ $EnergyStorageSystem \sqsubseteq \forall hasFrequency.Frequency$ $EnergyStorageSystem \sqsubseteq \exists hasOutput.BusNode$ $EnergyStorageSystem \sqsubseteq \forall hasPowerFactor.Powerfactor$ $EnergyStorageSystem \sqsubseteq \forall hasRatedChargeEfficiency.RatedChargeEfficiency$ $EnergyStorageSystem \sqsubseteq \forall hasRatedChargePower.RatedChargePower$ $EnergyStorageSystem \sqsubseteq \forall hasRatedCurrent.RatedCurrent$ $EnergyStorageSystem \sqsubseteq \forall hasRatedDischargeEfficiency.RatedDischargeEfficiency$
Lithium ion battery	$LithiumIonBattery \sqsubseteq \forall hasNumberOfCells.NumberOfCells$ $LithiumIonBattery \sqsubseteq \exists hasSubsystem.LithiumIonCell$
Hydrogen storage	$HydrogenStorage \sqsubseteq \exists hasSubsystem.(Electrolizer \sqcap FuelCell)$
Pump hydro	$Pumphydro \sqsubseteq \exists hasSubsystem.(ElectricalPump \sqcap Hydroelectricgenerator)$

Photo voltaic panel	$PhotovoltaicPanel \sqsubseteq \forall hasBaseTestingIrradiance.BaseTestingIrradiance$ $PhotovoltaicPanel \sqsubseteq \forall hasBaseTestingTeemperature.BaseTestingTemperature$ $PhotovoltaicPanel \sqsubseteq \forall hasNominalOperating-Temperature.NominalOperatingTemperature$ $PhotovoltaicPanel \sqsubseteq \forall hasPanelArea.PanelArea$ $PhotovoltaicPanel \sqsubseteq \forall hasPanelLength.PanelLength$ $PhotovoltaicPanel \sqsubseteq \forall hasPanelWidth.PanelWidth$ $PhotovoltaicPanel \sqsubseteq \forall hasRatedCurrent.RatedCurrent$ $PhotovoltaicPanel \sqsubseteq \forall hasRatedEfficiency.RatedEfficiency$ $PhotovoltaicPanel \sqsubseteq \forall hasRatedPower.RatedPower$ $PhotovoltaicPanel \sqsubseteq \forall hasRatedVoltage.RatedVolatge$ $PhotovoltaicPanel \sqsubseteq \forall hastemperatureCoeffOfPower.TemperatureCoefficientOfPower$

Power converter	$PowerConverter \sqsubseteq \forall hasActivePowerInput.InputActivePower$ $PowerConverter \sqsubseteq \forall hasActivePowerOutput.OutputActivePower$ $PowerConverter \sqsubseteq \forall hasCurrentInput.Current$ $PowerConverter \sqsubseteq \forall hasCurrentOutput.Current$ $PowerConverter \sqsubseteq \forall hasFlowDirection.FlowDirection$ $PowerConverter \sqsubseteq \forall hasFrequencyInput.InputFrequency$ $PowerConverter \sqsubseteq \forall hasFrequencyOutput.OutputFrequency$ $PowerConverter \sqsubseteq \exists hasInput.BusNode$ $PowerConverter \sqsubseteq \forall hasInputCurrenttype.InputCurrentType$ $PowerConverter \sqsubseteq \forall hasInputRatedCurrent.InputRatedCurrent$ $PowerConverter \sqsubseteq \forall hasInputRatedFrequency.InputRatedFrequency$ $PowerConverter \sqsubseteq \forall hasInputRatedVoltage.InputRatedVoltage$ $PowerConverter \sqsubseteq \exists hasOutput.BusNode$ $PowerConverter \sqsubseteq \forall hasOutputCurrentType.OutputCurrentType$ $PowerConverter \sqsubseteq \forall hasOutputRatedCurrent.OutputRatedCurrent$ $PowerConverter \sqsubseteq \forall hasOutputRatedFrequency.OutputRatedFrequency$

Power converter	$PowerConverter \sqsubseteq \forall hasOutputRatedVoltage.OutputRatedVoltage$ $PowerConverter \sqsubseteq \forall hasPowerFactorInput.InputPowerFactor$ $PowerConverter \sqsubseteq \forall hasPowerFactorOutput.OutputPowerFactor$ $PowerConverter \sqsubseteq \forall hasRatedEfficiency.RatedEfficiency$ $PowerConverter \sqsubseteq \forall hasRatedPower.RatedPower$ $PowerConverter \sqsubseteq \forall hasRatedPowerFactor.RatedPowerFactor$ $PowerConverter \sqsubseteq \forall hasReactivePowerInput.InputReactivePower$ $PowerConverter \sqsubseteq \forall hasReactivePowerOutput.OutputReactivePower$ $PowerConverter \sqsubseteq \forall hasVoltageInput.Voltage$ $PowerConverter \sqsubseteq \forall hasVoltageOutput.Voltage$ $PowerConverter \sqsubseteq \forall realizes.PowerConversion$
Transformer	$Transformer \sqsubseteq \forall hasEquivalentReactance.EquivalentReactance$ $Transformer \sqsubseteq \forall hasEquivalentResistance.EquivalentResistance$

Power Generator	$PowerGenerator \sqsubseteq \forall hasActivePowerGenerated.GeneratedActivePower$ $PowerGenerator \sqsubseteq \forall hasActualVoltage.Voltage$ $PowerGenarator \sqsubseteq \forall hasCurrent.Current$ $PowerGenarator \sqsubseteq \forall hasCurrentType.CurrentType$ $PowerGenarator \sqsubseteq \forall hasFrequency.Frequency$ $PowerGenarator \sqsubseteq \exists hasOutput.BusNode$ $PowerGenerator \sqsubseteq \forall hasPowerFactor.PowerFactor$ $PowerGenerator \sqsubseteq \forall hasRatedCurrent.RatedCurrent$ $PowerGenerator \sqsubseteq \forall hasRatedEfficiency.RatedEfficiency$ $PowerGenerator \sqsubseteq \forall hasRatedPower.RatedPower$ $PowerGenerator \sqsubseteq \forall hasRatedPowerFactor.RatedPowerFactor$ $PowerGenerator \sqsubseteq \forall hasRatedVoltage.RatedVoltage$ $PowerGenerator \sqsubseteq \forall hasReactivePowerGenerated.GeneratedReactivePower$ $PowerGenerator \sqsubseteq \exists hasShutdownCost.ShutdownCosts$ $PowerGenerator \sqsubseteq \exists hasStartupCost.StartupCosts$ $PowerGenerator \sqsubseteq \forall realizes.PowerGeneration$
Dispatchable Units	$DispatchableUnits \sqsubseteq \forall hasMinimumDownTime.MinimumDownTime$ $DispatchableUnits \sqsubseteq \forall hasMinimumUpTime.MinimumUpTime$ $DispatchableUnits \sqsubseteq \forall hasOperatingConstraint.OperatingConstraint$ $DispatchableUnits \sqsubseteq \forall hasShutDownTime.ShutDownTime$ $DispatchableUnits \sqsubseteq \forall hasStartupTime.StartupTime$

Photo Voltaic Generator	$PhotoVoltaicGenerator \sqsubseteq \exists hasCablingLosses.CablingLosses$ $PhotoVoltaicGenerator \sqsubseteq \exists hasDustLosses.DustLosses$ $PhotoVoltaicGenerator \sqsubseteq \exists hasHumidityLosses.HumidityLosses$ $PhotoVoltaicGenerator \sqsubseteq \forall hasInstallationAngle.InstallationAngles$ $PhotoVoltaicGenerator \sqsubseteq \forall hasNumberOfPanels.NumberOfPanels$ $PhotoVoltaicGenerator \sqsubseteq \forall hasOrientationAngle.Orientation$ $PhotoVoltaicGenerator \sqsubseteq \exists hasShadingLosses.ShadingLosses$ $PhotoVoltaicGenerator \sqsubseteq \forall hasTiltAngle.Tilt$ $PhotoVoltaicGenerator \sqsubseteq \exists hasSubsystem.(ElectricalLine \sqcap Photovoltaic \sqcap PowerCoverter)$

Power Load	$PowerLoad \sqsubseteq \forall hasActivePowerAbsorbed.AbsorbedActivePower$ $PowerLoad \sqsubseteq \forall hasActualVoltage.Voltage$ $PowerLoad \sqsubseteq \forall hasControllableStatus.ControllableStatus$ $PowerLoad \sqsubseteq \forall hasCriticalStatus.CriticalStatus$ $PowerLoad \sqsubseteq \forall hasCurrent.Current$ $PowerLoad \sqsubseteq \forall hasCurrentType.CurrentType$ $PowerLoad \sqsubseteq \forall hasFrequency.Frequency$ $PowerLoad \sqsubseteq \exists hasInput.BusNode$ $PowerLoad \sqsubseteq \forall hasPowerFactor.PowerFactor$ $PowerLoad \sqsubseteq \forall hasRatedCurrent.RatedCurrent$ $PowerLoad \sqsubseteq \forall hasRatedFrequency.RatedFrequency$ $PowerLoad \sqsubseteq \forall hasRatedPower.RatedPower$ $PowerLoad \sqsubseteq \forall hasRatedPowerFactor.RatedPowerFactor$ $PowerLoad \sqsubseteq \forall hasRatedVoltage.RatedVoltage$ $PowerLoad \sqsubseteq \forall hasReactivePowerAbsorbed.AbsorbedReactivePower$ $PowerLoad \sqsubseteq \forall realizes.PowerConsumption$
District	$District \sqsubseteq \forall hasDistricts.Building$
Substation	$Substation \sqsubseteq \neg(\exists hasShutDownCost.ShutDownCosts)$ $Substation \sqsubseteq \neg(\exists hasStartupCost.StartupCosts)$
Switch	$Switch \sqsubseteq \forall hasBreakerStatus.BreakerStatus$ $Switch \sqsubseteq \forall hasCurrentType.CurrentType$ $Switch \sqsubseteq \forall hasRatedCurrent.RatedCurrent$ $Switch \sqsubseteq \forall hasRatedVoltage.RatedVoltage$ $Switch \sqsubseteq \forall isConnectedTo.ElectricalLine$ $Switch \sqsubseteq \forall realizes.PowerFlow$

Voltage regulator	$VoltageRegulator \sqsubseteq \forall hasCurrentType.CurrentType$ $VoltageRegulator \sqsubseteq \forall hasFrequency.Frequency$ $VoltageRegulator \sqsubseteq \exists hasOutput.BusNode$ $VoltageRegulator \sqsubseteq \forall hasRatedCurrent.RatedCurrent$ $VoltageRegulator \sqsubseteq \forall hasRatedFrequency.RatedFrequency$ $VoltageRegulator \sqsubseteq \forall hasRatedPower.RatedPower$ $VoltageRegulator \sqsubseteq \forall hasRatedVoltage.RatedVoltage$ $VoltageRegulator \sqsubseteq \forall hasReactivePowerGenerated.GeneratedReactivePower$
Current	$Current \sqsubseteq \forall isComprisedOf.(CurrentAngle \sqcap CurrentMagnitude)$
Current angle	$CurrentAngle \sqsubseteq \exists hasBound.(MaximumCurrentAngle \sqcap MinimumCurrentAngle)$
Current magnitude	$CurrentMagnitude \sqsubseteq \exists hasBound.(MaximumCurrent \sqcap MinimumCurrent)$
Electrical network costs	$ElectricalNetworkCosts \sqsubseteq \exists hasDimension.amount\_of\_money$
Costs for systems realisation	$CostsForSystemsRealization \sqsubseteq \leqslant 2\, addsUp.\top$ $CostsForSystemsRealization \sqsubseteq \forall addsUp.(InstallationCostsForSystemsRealization \sqcup PurchaseCostsForSystemsRealization)$ $CostsForSystemsRealization \sqsubseteq \exists addsUp.InstallationCostsForSystemsRealization$ $CostsForSystemsRealization \sqsubseteq \exists addsUp.PurchaseCostsForSystemsRealization$

Direct manufacturing costs	$DirectManufacturingCosts \sqsubseteq \exists addsUp.Maintenance\&RepairCosts$ $DirectManufacturingCosts \sqsubseteq \exists addsUp.OperatingLaborCosts$ $DirectManufacturingCosts \sqsubseteq \exists addsUp.OperatingSupervisionCosts$ $DirectManufacturingCosts \sqsubseteq \exists addsUp.UtilityCosts$
Fixed capital investments	$FixedCapitalInvestments \sqsubseteq \leqslant 2\, addsUp.\top$ $FixedCapitalInvestments \sqsubseteq \forall addsUp.(ManufacturingFixedCapitalInvestmen \sqcup NonManufacturingFixedCapitalInvestment)$ $FixedCapitalInvestments \sqsubseteq \exists addsUp.ManufacturingFixedCapitalInvestment$ $FixedCapitalInvestments \sqsubseteq \exists addsUp.NonManufacturingFixedCapitalInvestment$
ManufacturingCosts	$ManufacturingCosts \sqsubseteq \exists addsUp.DirectManufacturingCosts$
Manufacturing fixed capital investment	$ManufacturingFixedCapitalInvestment \sqsubseteq \leqslant 4\; addsUp.\top$ $ManufacturingFixedCapitalInvestment \sqsubseteq \exists\; addsUp.CostsForLand$ $ManufacturingFixedCapitalInvestment \sqsubseteq \exists\; addsUp.CostsForSystemsRealization$ $ManufacturingFixedCapitalInvestment \sqsubseteq \exists\; addsUp.ServiceFacilityCosts$

ProductionCosts	$ProductionCosts \sqsubseteq \leqslant 2\, addsUp.\top$ $ProductionCosts \sqsubseteq \forall addsUp.(GeneralExpense \sqcup ManufacturingCosts)$ $ProductionCosts \sqsubseteq \exists addsUp.GeneralExpense$ $ProductionCosts \sqsubseteq \exists addsUp.ManufacturingCosts$
Purchase costs for systems realization	$PurchaseCostsForSystemsRealization \sqsubseteq \leqslant 3\, addsUp.\top$ $PurchaseCostsForSystemsRealization \sqsubseteq \exists addsUp.EquipmentCosts$ $PurchaseCostsForSystemsRealization \sqsubseteq \exists addsUp.InstrumentationCosts$
Total capital investment	$TotalCapitalInvestment \sqsubseteq \leqslant 2\, addsUp.\top$ $TotalCapitalInvestment \sqsubseteq \forall addsUp.(FixedCapitalInvestment \sqcup WorkingCapital)$ $TotalCapitalInvestment \sqsubseteq \exists addsUp.FixedCapitalInvestmentl$ $TotalCapitalInvestment \sqsubseteq \exists addsUp.WorkingCapital$
Total EN costs	$TotalENCosts \sqsubseteq \leqslant 2\, addsUp.\top$ $TotalENCosts \sqsubseteq \forall addsUp.(ProductionCosts \sqcup TotalCapitalInvestment)$ $TotalENCosts \sqsubseteq \exists addsUp.ProductionCosts$ $TotalENCosts \sqsubseteq \exists addsUp.TotalCapitalInvestment$
Generated reactive power	$GeneratedReactivePower \sqsubseteq \exists hasBound.(MaximumReactivePower \sqcap MinimumReactivePower)$
InputReactivePower	$InputReactivePower \sqsubseteq \exists hasBound.(MaximumReactivePower \sqcap MinimumReactivePower)$
OutputReactivePower	$OutputReactivePower \sqsubseteq \exists hasBound.(MaximumReactivePower \sqcap MinimumReactivePower$
VoltageAngle	$VoltageAngle \sqsubseteq \exists hasBound.(MaximumVoltageAngle \sqcap MinimumVoltageAngle$

VoltageMagnitude	$VoltageMagnitude \sqsubseteq \exists hasBound.(MaximumVoltageMagnitude \sqcap MinimumVoltageMagnitude$
Voltage	$Voltage \sqsubseteq \exists isComprisedOf.(VoltageAngle \sqcap VoltageMagnitude)$
Economic Performance	$EconomicPerformance \sqsubseteq \exists hasProperty.Earnings$ $EconomicPerformance \sqsubseteq \exists hasProperty.ElectricalNetworkCosts$ $EconomicPerformance \sqsubseteq \exists representsPerformanceOf.PowerSystem$
Model	$Model \sqsubseteq \forall models.System$
Mathematical model	$MathematicalModel \sqsubseteq \forall hasCoupling.Coupling$ $MathematicalModel \sqsubseteq \forall hasModelPort.ModelPort$ $MathematicalModel \sqsubseteq \forall hasModelVariable.ModelVariable$
Power system model	$PowerSystemModel \sqsubseteq (= 1\, hasModellingPrinciple.\top)$ $PowerSystemModel \sqsubseteq \forall hasModellingPrinciple.ModellingPrinciple$ $PowerSystemModel \sqsubseteq \exists models.PowerSystem$
Electrical domain law	$ElectricalDomainLaw \sqsubseteq \leqslant 1\, associatedWith.\top$ $ElectricalDomainLaw \sqsubseteq \forall isDirectSubsystemOf.PowerSystemModel$
Forecast load model agent	$ForecastLoadModelAgent \sqsubseteq \exists hasModelVariable.ForecastLoad$
Forecast price model agent	$ForecastPriceModelAgent \sqsubseteq \exists hasModelVariable.ForecastPrices$

Power dispatch model agent	$PowerDispatchModelAgent \sqsubseteq \exists hasModelVariable.$ $(ForecastLoad \sqcup ForecastPrices \sqcup ForecastSolar-$ $Generation \sqcup PmaxBattery \sqcup PMaxGridmetered \sqcup$ $PMinBattery \sqcup PMinGridMetered \sqcup Predicted-$ $GeneratedActivePower \sqcup PredictedMeterd-$ $PowerFlow \sqcup RatedEff \sqcup Sampling \sqcup SoC \sqcup SoCFinal$ $\sqcup SoCMax \sqcup SocMin)$
Power flow model agent	$PowerFlowModelAgent \sqsubseteq \exists hasModelVariable.$ $(BranchInputVariable \sqcup Branchparameter$ $\sqcup BranchStateVariable \sqcup BusInputVariable \sqcup$ $BusNodeparameter \sqcup BusStateVariable \sqcup Generator-$ $InputVariable \sqcup GeneratorParameter \sqcup$ $GeneratorStateVariable)$
Property model	$PropertyModel \sqsubseteq \forall isDirectSubsystemOf.$ $PowerSystemModel$
Solar generator model agent	$SolarGeneratorModel \sqsubseteq \forall hasModelVariable.$ $ForecastSolarGeneration$

Table 9: *RI and CI axioms for given role names in OntoPowSys.*

Role name	**Axioms**
hasCableType	$hasCableType \sqsubseteq hasCharacteristic$
hasCapacity	$hasCapacity \sqsubseteq hasCharacteristic$
hasModelingPrinciple	$hasModelingPrinciple \sqsubseteq hasCharacteristic$
hasEnergySource	$hasModelingPrinciple \sqsubseteq hasCharacteristic$ $\exists hasEnergySource.\top \sqsubseteq PowerGenerator$ $\top \sqsubseteq \forall hasEnergySource.EnergySource$
hasInsulationMaterial	$hasInsulationMaterial \sqsubseteq inter-objectRelation$ $\exists hasInsulationMaterial.\top \sqsubseteq UndergroundCable$ $\top \sqsubseteq \forall hasInsulationMaterial.Material$

hasBound	$hasBound \sqsubseteq inter-objectRelation$ $\exists hasBound.\top \sqsubseteq Property$ $\top \sqsubseteq \forall hasBound.ConstantProperty$
hasCost	$hasCost \sqsubseteq inter-objectRelation$ $\top \sqsubseteq \leqslant 1 hasCost^{-}.\top$
hasDegradationCost	$hasDegradationCost \sqsubseteq hasCost$
hasDirectCost	$hasDirectCost \sqsubseteq hasCost$
hasEnvironmentalCost	$hasEnvironmentalCost \sqsubseteq hasCost$
hasEquipmentCost	$hasEquipmentCost \sqsubseteq hasCost$
hasFuelCost	$hasFuelCost \sqsubseteq hasCost$
hasInstallationCost	$hasInstallationCost \sqsubseteq hasCost$
hasLaborCost	$hasLaborCost \sqsubseteq hasCost$
hasMaintenanceCost	$hasMaintenanceCost \sqsubseteq hasCost$
hasFixedMaintenanceCost	$hasFixedMaintenanceCost \sqsubseteq hasMaintenanceCost$
hasManufacturingCost	$hasManufacturingCost \sqsubseteq hasCost$
hasShutdownCost	$hasShutdownCost \sqsubseteq hasCost$
hasStartupCost	$hasStartupCost \sqsubseteq hasCost$
hasUtilityCost	$hasUtilityCost \sqsubseteq hasCost$
addsUp	$addsUp \sqsubseteq hasDirectPart$ $\exists addsUp.\top \sqsubseteq AccumulatedCosts$ $\top \sqsubseteq \forall addsUp.ElectricalNetworkCosts$
hasAngle	$hasAngle \sqsubseteq isComposedOf$ $\exists hasAngle\top \sqsubseteq Property$ $\top \sqsubseteq \forall hasAngle.ScalarQuantity$
hasMagnitude	$hasMagnitude \sqsubseteq isComposedOf$ $\exists hasMagnitude\top \sqsubseteq Property$ $\top \sqsubseteq \forall hasMagnitude.ScalarQuantity$
hasPrice	$hasPrice \sqsubseteq inter-objectRelation$
hasActivePowerAbsorbed	$hasActivePowerAbsorbed \sqsubseteq hasProperty$ $\top \sqsubseteq \forall hasActivePowerAbsorbed.AbsorbedActivePower$

hasGeneratedActivePower	$hasGeneratedActivePower \sqsubseteq hasProperty$ $\top \sqsubseteq \forall hasGeneratedActivePower.GeneratedActivePower$
hasActivePowerInput	$hasActivePowerInput \sqsubseteq hasProperty$ $\top \sqsubseteq \forall hasActivePowerAbsorbed.InputActivePower$
hasActivePowerLoss	$hasActivePowerLoss \sqsubseteq hasProperty$ $\top \sqsubseteq \forall hasActivePowerLoss.ActivePowerLosses$
hasActivePowerOutput	$hasActivePowerOutput \sqsubseteq hasProperty$ $\top \sqsubseteq \forall hasActivePowerOutput.OutputActivePower$
hasActualVoltage	$hasActualVoltage \sqsubseteq hasProperty$
hasActualHigherVoltage	$hasActualHigherVoltage \sqsubseteq hasActualVoltage$
hasActualLowerVoltage	$hasActualLowerVoltage \sqsubseteq hasActualVoltage$
hasVoltageInput	$hasVoltageInput \sqsubseteq hasActualVoltage$
hasVoltageOutput	$hasVoltageOutput \sqsubseteq hasActualVoltage$
hasBaseTestingIrradiance	$hasBaseTestingIrradiance \sqsubseteq hasProperty$
hasBaseTestingTemperature	$hasBaseTestingTemperature \sqsubseteq hasProperty$
hasBreakerStatus	$hasBreakerStatus \sqsubseteq hasProperty$ $\top \sqsubseteq \forall hasBreakerStatus.BreakerStatus$
hasCableGeometry	$hasCableGeometry \sqsubseteq hasProperty$ $\top \sqsubseteq \forall hasCableGeometry.CableGeometry$
hasControllableStatus	$hasControllableStatus \sqsubseteq hasProperty$
hasCriticalStatus	$hasCriticalStatus \sqsubseteq hasProperty$
hasCurrent	$hasCurrent \sqsubseteq hasProperty$
hasCurrentInput	$hasCurrentInput \sqsubseteq hasCurrent$
hasCurrentOutput	$hasCurrentOutput \sqsubseteq hasCurrent$
hasCurrentType	$hasCurrentType \sqsubseteq hasProperty$
hasInputCurrentType	$hasInputCurrentType \sqsubseteq hasCurrentType$
hasOutputCurrentType	$hasOutputCurrentType \sqsubseteq hasCurrentType$
hasEfficiency	$hasEfficiency \sqsubseteq hasProperty$

hasEquivalentReactance	$hasEquivalentReactance \sqsubseteq hasProperty$
hasEquivalentResistance	$hasEquivalentResistance \sqsubseteq hasProperty$
hasFlowDirection	$hasFlowDirection \sqsubseteq hasProperty$ $\exists hasFlowDirection.\top \sqsubseteq PowerConverter$ $\top \sqsubseteq \forall hasFlowDirection.FlowDirection$
hasFrequency	$hasFrequency \sqsubseteq hasProperty$ $\top \sqsubseteq \forall hasFrequency.Frequency$
hasFrequencyInput	$hasFrequencyInput \sqsubseteq hasFrequency$ $\top \sqsubseteq \forall hasFrequencyInput.InputFrequency$
hasFrequencyOutput	$hasFrequencyOuput \sqsubseteq hasFrequency$ $\top \sqsubseteq \forall hasFrequencyOutput.OutputFrequency$
hasInstallationAngle	$hasInstallationAngle \sqsubseteq hasProperty$
hasOrientationAngle	$hasOrientationAngle \sqsubseteq hasInstallationAngle$
hasTiltAngle	$hasInstallationAngle \sqsubseteq hasInstallationAngle$
hasLineGeometry	$\sqsubseteq hasProperty$
hasModelPort	$hasModelPort \sqsubseteq hasProperty$ $\exists hasModelPort.\top \sqsubseteq MathematicalModel$ $\top \sqsubseteq \forall hasModelPort.ModelPort$
hasModelVariable	$hasModelVariable \sqsubseteq hasProperty$ $\exists hasModelVariable.\top \sqsubseteq MathematicalModel$ $\top \sqsubseteq \forall hasModelVariable.ModelVariable$
hasNominalOperatingTempera	$hasNominalOperatingTemperature \sqsubseteq hasProperty$
hasNumberOfCells	$hasNumberOfCells \sqsubseteq hasProperty$
hasNumberOfPanels	$hasNumberOfPanels \sqsubseteq hasProperty$
hasOperatingConstraint	$hasOperatingConstraint \sqsubseteq hasProperty$
hasMinimumDownTime	$hasMinimumDownTime \sqsubseteq hasOperatingConstraint$
hasMinimumUpTime	$hasMinimumUpTime \sqsubseteq hasOperatingConstraint$
hasShutdownTime	$hasShutdownTime \sqsubseteq hasOperatingConstraint$
hasStartupTime	$hasStartupTime \sqsubseteq hasOperatingConstraint$

hasPanelArea	$hasPanelArea \sqsubseteq hasProperty$
hasPanelLength	$hasPanelLength \sqsubseteq hasProperty$
hasPanelWidth	$hasPanelWidth \sqsubseteq hasProperty$
hasPowerFactor	$hasPowerFactor \sqsubseteq hasProperty$ $\top \sqsubseteq \forall hasPowerFactor.PowerFactor$
hasPowerFactorInput	$hasPowerFactorInput \sqsubseteq hasPowerFactorInput$ $\top \sqsubseteq \forall hasPowerFactorInput.InputPowerFactor$
hasPowerFactorOutput	$hasPowerFactorInput \sqsubseteq hasPowerFactorOutput$ $\top \sqsubseteq \forall hasPowerFactorOutput.OutputPowerFactor$
hasPowerInjection	$hasPowerInjection \sqsubseteq hasProperty$
hasActivePowerInjection	$hasActivePowerInjecton \sqsubseteq hasPowerInjection$ $\top \sqsubseteq \forall hasActivePowerInjection.ActivePowerbalance$
hasReactivePowerInjection	$hasReactivePowerInjection \sqsubseteq hasPowerInjection$ $\top \sqsubseteq \forall hasReactivePowerInjection.ActivePowerbalance$
hasInputRatedCurrent	$hasRatedCurrent \sqsubseteq hasRatedCurrent$
hasOutputRatedCurrent	$hasOutputRatedCurrent \sqsubseteq hasRatedCurrent$
hasRatedCurrent	$hasRatedCurrent \sqsubseteq hasProperty$
hasRatedEfficiency	$hasRatedEfficiency \sqsubseteq hasProperty$
hasRatedChargeEfficiency	$hasRatedChargeEfficiency \sqsubseteq hasRatedEfficiency$
hasRatedDischargeEfficiency	$hasRatedDischargeEfficiency \sqsubseteq hasRatedEfficiency$
hasRoundTripEfficiency	$hasRoundTripEfficiency \sqsubseteq hasRatedEfficiency$
hasRatedEnergyCapacity	$hasRatedEnergyCapacity \sqsubseteq hasProperty$
hasRatedFrequency	$hasRatedFrequency \sqsubseteq hasProperty$
hasInputRatedFrequency	$hasInputRatedFrequency \sqsubseteq hasRatedFrequency$
hasOutputRatedFrequency	$hasOutputRatedFrequency \sqsubseteq hasRatedFrequency$
hasRatedPower	$hasRatedPower \sqsubseteq hasProperty$
hasRatedChargePower	$hasRatedChargePower \sqsubseteq hasRatedpower$
hasRatedDischargePower	$hasRatedDischargePower \sqsubseteq hasRatedPower$
hasRatedPowerFactor	$hasRatedPowerFactor \sqsubseteq hasProperty$

hasRatedVoltage	$hasRatedVoltage \sqsubseteq hasProperty$
hasInputRatedVoltage	$hasInputRatedVoltage \sqsubseteq hasRatedVoltage$
hasOuputRatedVoltager	$hasOutputRatedVoltage \sqsubseteq hasRatedVoltage$
hasReactance	$hasReactance \sqsubseteq hasProperty$
hasReactivePowerAbsorbed	$hasReactivePowerAbsorbed \sqsubseteq hasProperty$
hasReactivePowerGenerated	$hasReactivePowerGenerated \sqsubseteq hasProperty$
hasReactivePowerInput	$hasReactivePowerInput \sqsubseteq hasProperty$ $\top \sqsubseteq \forall hasReactivePowerInput.InputReactivePower$
hasReactivePowerLoss	$hasReactivePowerLoss \sqsubseteq hasProperty$ $\top \sqsubseteq \forall hasReactivePowerLoss.ReactivePowerLosses$
hasReactivePowerOutput	$hasReactivePowerOutput \sqsubseteq hasProperty$ $\top \sqsubseteq \forall hasReactivePowerOutput.OutputReactivePower$
hasRegulatedBus	$hasRegulatedBus \sqsubseteq hasProperty$ $\exists hasRegulatedBus.\top \sqsubseteq PowerGenerator$
hasResistance	$hasResistance \sqsubseteq hasProperty$
hasSamplingTime	$hasSamplingTime \sqsubseteq hasProperty$
hasStateOfCharge	$hasStateOfCharge \sqsubseteq hasProperty$
hasSusceptance	$hasSusceptance \sqsubseteq hasProperty$
hasSystemLosses	$hasSystemLosses \sqsubseteq hasProperty$
hasCablingLosses	$hasCablingLosses \sqsubseteq hasSystemLosses$ $\exists hasCablingLosses.\top \sqsubseteq PhotovoltaicGenerator$ $\top \sqsubseteq \forall hasCablingLosses.CablingLosses$
hasDustLosses	$hasDustLosses \sqsubseteq hasSystemLosses$ $\exists hasCablingLosses.\top \sqsubseteq PhotovoltaicGenerator$ $\top \sqsubseteq \forall hasDustLosses.DustLosses$
hasHumidityLosses	$hasHumidityLosses \sqsubseteq hasSystemLosses$ $\exists hasHumidityLosses.\top \sqsubseteq PhotovoltaicGenerator$ $\top \sqsubseteq \forall hasHumidityLosses.HumidityLosses$

hasInverterLosses	$hasInverterLosses \sqsubseteq hasSystemLosses$ $\exists hasInverterLosses.\top \sqsubseteq PhotovoltaicGenerator$ $\top \sqsubseteq \forall hasInverterLosses.InverterLosses$
hasShadingLosses	$hasShadingLosses \sqsubseteq hasSystemLosses$ $\exists hasShadingLosses.\top \sqsubseteq PhotovoltaicGenerator$ $\top \sqsubseteq \forall hasShadingLosses.ShadingLosses$
hasTemperatureLosses	$hasTemperatureLosses \sqsubseteq hasSystemLosses$ $\exists hasTemperatureLosses.\top \sqsubseteq PhotovoltaicGenerator$ $\top \sqsubseteq \forall hasTemperatureLosses.TemepratureLosses$
hasWeakIrrLosses	$hasWeakIrrLosses \sqsubseteq hasSystemLosses$ $\exists hasWeakIrrLosses.\top \sqsubseteq PhotovoltaicGenerator$ $\top \sqsubseteq \forall hasWeakIrrLosses.WeakIrradiationLosses$
hasTemperatureCoeffOfPower	$hasTemperatureCoeffOfPower \sqsubseteq hasProperty$
hasVoltageAngle	$hasVoltageAngle \sqsubseteq hasProperty$
hasHigherVoltageAngle	$hasHigherVoltageAngle \sqsubseteq hasVoltageAngle$
hasLowerVoltageAngle	$hasLowerVoltageAngle \sqsubseteq hasVoltageAngle$
hasRevenue	$hasRevenue \sqsubseteq inter-ObjectRelation$
isAssociatedWith	$isAssociatedWith \sqsubseteq object-featureRelation$

D. OntoEIP axioms

Table 10: *CI axioms in OntoEIP.*

Term name	**Axioms**
Stirred tank	$StirredTank \sqsubseteq Vessel$ $StirredTank \sqsubseteq \exists hasHeatDuty.MassTransferCoefficient$
Mass transfer coefficient	$MassTransferCoefficient \sqsubseteq \exists hasValue.ScalarValue$
Temperature	$Temperature \sqsubseteq ThermodynamicStateProperty$ $Temperature \sqsubseteq \exists hasDimension.PhysicalDimension$ $Temperature \sqsubseteq \exists hasValue.ScalarValue$

Scalar value	$ScalarValue \sqsubseteq \exists hasUnitOfMeasure \leqslant 1 .UnitOfMeasure$ $ScalarValue \sqsubseteq \exists numericalValue = 1.(rdfs : literal)$
Composite system	$CompositeSystem \sqsubseteq \exists hasSubsystem.System$ $CompositeSystem \sqsubseteq \exists hasAspectSystem.System$

Table 11: *OntoEIP RI and CI axioms used for inferring new individual assertions.*

Role name	**Axioms**
hasProperty	$hasProperty \sqsubseteq inter-objectRelation$ $hasProperty \equiv isPropertyOf^{-}$ $\top \sqsubseteq \leqslant 1\, hasProperty^{-}$ $\exists hasProperty.\top \sqsubseteq System$ $\top \sqsubseteq \forall hasProperty.Property$
hasValue	$hasValue \sqsubseteq inter-objectRelation$ $hasValue \equiv isValueOf^{-}$ $\top \sqsubseteq \leqslant 1\, hasValue^{-}$ $\exists hasValue.\top \sqsubseteq Property$ $\top \sqsubseteq \forall hasValue.Value$
hasPart	$hasPart \sqsubseteq inter-objectRelation$ $hasPart \equiv isPartOf^{-}$ $\exists hasPart.\top \sqsubseteq Object$ $\top \sqsubseteq \forall hasValue.Object$
hasDirectPart	$hasDirectPart \sqsubseteq hasPart$ $hasDirectPart \equiv isDirectPartOf^{-}$
isComposedOf	$isComposedOf \sqsubseteq hasDirectPart$ $hasPart \equiv isExclusivelyPartOf^{-}$
hasAspectSystem	$hasAspectSystem \sqsubseteq isComposedOfSubsystem$ $hasPart \equiv representsAspectOf^{-}$ $\top \sqsubseteq \forall hasAspectSystem.AspectSystem$

hasShapeRepresentation	$hasShapeRepresentation \sqsubseteq isComposedOfSubsystem$ $hasShapeRepresentation \equiv representsShapeOf^{-}$ $\top \sqsubseteq \forall hasShapeRepresentation.Solid$
hasInput	$hasInput \sqsubseteq isSuccessorOf$ $hasInput \equiv enters^{-}$ $\exists hasInput.\top \sqsubseteq Node$ $\top \sqsubseteq \forall hasInput.DirectedArc$
hasOutput	$hasOutput \sqsubseteq isPredecessorOf$ $hasOutput \equiv leaves^{-}$ $\exists hasOutput.\top \sqsubseteq Node$ $\top \sqsubseteq \forall hasOutput.DirectedArc$
isConnectedTo	$isConnectedTo \sqsubseteq inter-ObjectRelation$ $isConnectedTo \circ isConnectedTo \sqsubseteq isConnectedTo$ $isConnectedTo^{-} \sqsubseteq isConnectedTo$
isSuccessorOf	$isSuccessorOf \sqsubseteq isConnectedTo$ $isSuccessorOf \equiv isPredecessorOf^{-}$ $isSuccessorOf \circ isSuccessorOf \sqsubseteq isSuccessorOf$ $\exists isSuccessorOf.\top \sqsubseteq (DirectedArc \sqcup Node)$ $\top \sqsubseteq \forall hasInput.(DirectedArc \sqcup Node)$
haslength	$haslength \sqsubseteq hasProperty$ $\exists haslength.\top \sqsubseteq (Solid \sqcup Surface)$ $\top \sqsubseteq \forall haslength.(Diameter \sqcup EdgeLength \sqcup Height-$ $\sqcup Radius)$
isDirectlyRelatedTo	$isDirectlyRelatedTo^{-} \sqsubseteq isDirectlyRelatedTo$ $isDirectlyRelatedTo \equiv isDirectlyRelatedTo^{-}$
isRelatedTo	$isRelatedTo \sqsubseteq inter-objectRelation$ $isRelatedTo\ \ o\ \ isRelatedTo \sqsubseteq isRelatedTo$ $isRelatedTo^{-} \sqsubseteq isRelatedTo$ $\exists isRelatedTo.\top \sqsubseteq System$ $\top \sqsubseteq \forall isRelatedTo.System$

isRealizedBy	$isRealizedBy \sqsubseteq isDirectlyRelatedTo$ $isRealizedBy \equiv realizes^-$ $\exists isRelatedTo.\top \sqsubseteq SystemFunction$ $\top \sqsubseteq \forall isRelatedTo.SystemRealization$
isModeledBy	$isModeledBy \sqsubseteq isDirectlyRelatedTo$ $isModeledBy \equiv models^-$ $\exists isRelatedTo.\top \sqsubseteq System$ $\top \sqsubseteq \forall isModeledBy.Model$
hasSubsystem	$hasSubsystem \sqsubseteq isRelatedTo$ $hasSubsystem \equiv isSubsystemOf^-$ $\exists hasSubsystem.\top \sqsubseteq SuperSystem$ $\top \sqsubseteq \forall hasSubsystem.System$ $hasSubsystem \circ hasSubsystem \sqsubseteq hasSubsystem$

References

[1] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, and J. T. Eppig. Gene ontology: Tool for the unification of biology. *Nature Genetics*, 25(1):25, 2000. doi:10.1038/75556.

[2] D. Bonino and L. De Russis. DogOnt as a viable seed for semantic modeling of AEC/FM. *Semantic Web*, (Preprint):1–18, 2018. doi:10.3233/SW-180295.

[3] C. Daraio, M. Lenzerini, C. Leporelli, P. Naggar, A. Bonaccorsi, and A. Bartolucci. The advantages of an Ontology-Based Data Management approach: openness, interoperability and data quality. *Scientometrics*, 108(1):441–455, July 2016. doi:10.1007/s11192-016-1913-6.

[4] A. Devanand, M. Kraft, and I. Karimi. Optimal site selection for modular nuclear power plants. *Computers and Chemical Engineering*, 125, 2019. doi:10.1016/j.compchemeng.2019.03.024.

[5] M. Eid, R. Liscano, and A. El Saddik. A novel ontology for sensor networks data. In *IEEE International Conference on Computational Intelligence for Measurement Systems and Applications*, pages 75–79. IEEE, 2006.

[6] M. Fernández-López, A. Gómez-Pérez, and N. Juristo. Methontology: from ontological art towards ontological engineering. In *Proceedings of the Ontological Engineering AAAI-97 Spring Symposium Series*. American Asociation for Artificial Intelligence, 1997. URL `http://oa.upm.es/5484`.

[7] O. Givehchi, K. Landsdorf, P. Simoens, and A. W. Colombo. Interoperability for Industrial Cyber-Physical Systems: An Approach for Legacy Systems. *IEEE Transactions on Industrial Informatics*, 13(6):3370–3378, Dec 2017. doi:10.1109/TII.2017.2740434.

[8] B. Glimm, I. Horrocks, B. Motik, G. Stoilos, and Z. Wang. HermiT: An OWL 2 Reasoner. *Journal of Automated Reasoning*, 53(3):245–269, Oct 2014. doi:10.1007/s10817-014-9305-1.

[9] T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993. doi:10.1006/knac.1993.1008.

[10] P. Hitzler, M. Krötzsch, B. Parsia, P. Patel-Schneider, and S. Rudolph. OWL 2 Web Ontology Language Primer. *W3C recommendation*, 2012. URL `https://www.w3.org/TR/owl2-primer/`.

[11] H. Hlomani and D. Stacey. Approaches, methods, metrics, measures, and subjectivity in ontology evaluation: A survey. *Semantic Web Journal*, 1(5):1–11, 2014. URL `http://www.semantic-web-journal.net/system/files/swj657.pdf`.

[12] Y. Huang and X. Zhou. Knowledge model for electric power big data based on ontology and semantic web. *CSEE Journal of Power and Energy Systems*, 1(1): 19–27, 2015. doi:10.17775/CSEEJPES.2015.00003.

[13] G. Kourtis, E. Kavakli, and R. Sakellariou. A Rule-Based Approach Founded on Description Logics for Industry 4.0 Smart Factories. *IEEE Transactions on Industrial Informatics*, pages 1–1, 2019. doi:10.1109/TII.2019.2916622.

[14] E. A. Lee. Cyber physical systems: Design challenges. In *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)*, pages 363–369. IEEE, 2008.

[15] P. Leitão, V. Mařík, and P. Vrba. Past, Present, and Future of Industrial Agent Applications. *IEEE Transactions on Industrial Informatics*, 9(4):2360–2372, Nov 2013. doi:10.1109/TII.2012.2222034.

[16] Lida. Xu, W. He, and S. Li. Internet of things in industries: A survey. *IEEE Transactions on Industrial Informatics*, 10(4):2233–2243, Nov 2014. doi:10.1109/TII.2014.2300753.

[17] J. S. Luciano, B. Andersson, C. Batchelor, O. Bodenreider, T. Clark, C. K. Denney, C. Domarew, T. Gambet, L. Harland, A. Jentzsch, V. Kashyap, P. Kos, J. Kozlovsky, T. Lebo, S. M. Marshall, J. P. McCusker, D. L. McGuinness, C. Ogbuji, E. Pichler, R. L. Powers, E. Prud'hommeaux, M. Samwald, L. Schriml, P. J. Tonellato, P. L. Whetzel, J. Zhao, S. Stephens, and M. Dumontier. The Translational Medicine Ontology and Knowledge Base: driving personalized medicine by bridging the gap between bench and bedside. *Journal of Biomedical Semantics*, 2(Suppl 2):S1, 2011. doi:10.1186/2041-1480-2-S2-S1.

[18] S. A. Martin, K. A. Weitz, R. A. Cushman, A. Sharma, R. C. Lindrooth, and S. R. Moran. Eco-Industrial Parks: A case study and analysis of economic, environmental, technical, and regulatory issues. Office of Policy, Planning and Evaluation, USEPA, Washington, DC, USA, 1996. URL `https://pdxscholar.library.pdx.edu/metropolitianstudies/110/`.

[19] D. L. McGuinness and F. Van Harmelen. OWL web ontology language overview. *W3C recommendation*, 10(10), 2004. URL `http://www.w3.org/TR/2004/REC-owl-features-20040210/`.

[20] J. Morbach, A. Yang, and W. Marquardt. OntoCAPE-A large-scale ontology for chemical process engineering. *Engineering Applications of Artificial Intelligence*, 20(2):147–161, 2007. doi:10.1016/j.engappai.2006.06.010.

[21] J. Morbach, A. Wiesner, and W. Marquardt. OntoCAPE-A (re)usable ontology for computer-aided process engineering. *Computers and Chemical Engineering*, 33(10): 1546–1556, 2009. doi:10.1016/j.compchemeng.2009.01.019.

[22] M. A. Musen and Protégé Team. The Protégé Project: A Look Back and a Look Forward. *AI Matters*, 1(4):4–12, 2015. doi:10.1145/2757001.2757003.

[23] Y. Pradeep, S. A. Khaparde, and R. K. Joshi. High level event ontology for multiarea power system. *IEEE Transactions on Smart Grid*, 3(1):193–202, 2012. doi:10.1109/TSG.2011.2173508.

[24] J. Puttonen, A. Lobov, and J. L. Martinez Lastra. Semantics-Based Composition of Factory Automation Processes Encapsulated by Web Services. *IEEE Transactions on Industrial Informatics*, 9(4):2349–2359, Nov 2013. doi:10.1109/TII.2012.2220554.

[25] S. Rudolph. Foundations of description logics. *Lecture Notes in Computer Science*, 6848 LNCS:76–136, 2011. doi:10.1007/978-3-642-23032-5_2.

[26] G. Santos, T. Pinto, H. Morais, T. M. Sousa, I. F. Pereira, R. Fernandes, I. Praça, and Z. Vale. Multi-agent simulation of competitive electricity markets: Autonomous systems cooperation for European market modeling. *Energy Conversion and Management*, 99:387–399, 2015. doi:10.1016/j.enconman.2015.04.042.

[27] G. F. Schneider, P. Pauwels, and S. Steiger. Ontology-based modeling of control logic in building automation systems. *IEEE Transactions on Industrial Informatics*, 13(6):3350–3360, Dec 2017. doi:10.1109/TII.2017.2743221.

[28] G. Schreiber, B. Wielinga, and W. Jansweijer. The KACTUS View on the O Word. In *IJCAI Workshop on Basic Ontological Issues in Knowledge Sharing*, page 10, 1995. URL `https://www.researchgate.net/profile/Bob_Wielinga/publication/2942056_The_KACTUS_View_on_the_'O'_Word/links/0fcfd508a9cdbdece5000000/The-KACTUS-View-on-the-O-Word.pdf`.

[29] B. Swartout, R. Patil, K. Knight, and T. Russ. Toward distributed use of large-scale ontologies. In *Proceedings of the Tenth Workshop on Knowledge Acquisition for Knowledge-Based Systems*, pages 138–148, 1996. URL `https://www.aaai.org/Papers/Symposia/Spring/1997/SS-97-06/SS97-06-018.pdf`.

[30] T. Tudor, E. Adam, and M. Bates. Drivers and limitations for the successful development and functioning of EIPs (Eco-Industrial Parks): A literature review. *Ecological Economics*, 61(2):199–207, 2007. doi:10.1016/j.ecolecon.2006.10.010.

[31] M. Wollschlaeger, T. Sauter, and J. Jasperneite. The future of industrial communication: Automation networks in the era of the internet of things and Industry 4.0. *IEEE Industrial Electronics Magazine*, 11(1):17–27, 2017. URL `https://www.hs-owl.de/init/uploads/tx_initdb/IEEEMagazine.pdf`.

[32] C. Zhang, L. Zhou, P. Chhabra, S. S. Garud, K. Aditya, A. Romagnoli, G. Comodi, F. Dal Magro, A. Meneghetti, and M. Kraft. A novel methodology for the design of waste heat recovery network in Eco-Industrial Park using techno-economic analysis and multi-objective optimization. *Applied Energy*, 184:88–102, dec 2016. doi:10.1016/j.apenergy.2016.10.016.

[33] C. Zhang, A. Romagnoli, L. Zhou, and M. Kraft. Knowledge management of Eco-Industrial Park for efficient energy utilization through ontology-based approach. *Applied Energy*, 204:1412–1421, 2017. doi:10.1016/j.apenergy.2017.03.130.

[34] L. Zhou, M. Pan, J. J. Sikorski, S. Garud, L. K. Aditya, M. J. Kleinelanghorst, I. A. Karimi, and M. Kraft. Towards an ontological infrastructure for chemical process simulation and optimization in the context of Eco-Industrial Parks. *Applied Energy*, 204:1284–1298, 2017. doi:10.1016/j.apenergy.2017.05.002.