

Intelligence Distribution for Data Processing in Smart Grids: A Semantic Approach

Juan Carlos Nieves^a, Angelina Espinoza^b, Yoseba K. Peña^c, Mariano Ortega de Mues^d, Aitor Peña^c

^a*Department of Computing Sciences, Umeå University, SE-901 87, Umeå, Sweden*

^b*Electrical Engineering Department, Unidad Iztapalapa, Universidad Autónoma Metropolitana, Av. San Rafael Atlixco 186, México City, 09340, México*

^c*DeustoTech, University of Deusto, Bilbao, Basque Country, Spain*

^d*INDRA Sistemas S.A., Energy Market Department, Avda.de Bruselas 35, E-28108, Madrid, Spain*

Abstract

The smart grid vision demands both *syntactic* interoperability in order to physically be able to interchange data and *semantic* interoperability to properly understand and interpret its meaning. The IEC and the EPRI have backed to this end the harmonization of two widely-used industrial standards, the CIM and the IEC 61850, as the global unified ontology in the smart grid scenario. Still, persisting such a huge general ontology in each and every one of the members of a distributed system is neither practical nor feasible.

Moreover, the smart grid will be a heterogeneous conglomerate of legacy and upcoming architectures that will require first the possibility of representing all the existing assets in the power network as well as new unknown ones, and second, the collaboration of different entities of the system in order to deploy complex activities. Finally, the smart grid presents diverse time span requirements, such as real-time, and all of them must be addressed efficiently but use resources sparingly.

Against this background, we put forward an architecture of intelligent nodes spread all over the smart grid structure. Each intelligent node only has *a profile* of the global ontology. Moreover, adding reasoning abili-

Email addresses: jcnieves@cs.umu.se (Juan Carlos Nieves),
angelina.espinoza.limon@gmail.com (Angelina Espinoza),
yoseba.pena@deusto.es (Yoseba K. Peña), mortega@indra.es (Mariano Ortega de Mues), aitor.pena@deusto.es (Aitor Peña)

ties, we achieve simultaneously the required intelligence distribution and local decision making. Furthermore, we address the aforementioned real-time and quasi-real-time requirements by integrating stream data processing tools within the intelligent node. Combined with the knowledge base profile and the reasoning capability, our intelligent architecture supports semantic stream reasoning. We have illustrated the feasibility of this approach with a prototype composed of three substations and the description of several complex activities involving a number of different entities of the smart grid. Moreover, we have also addressed the potential extension of the unified ontology.

Keywords: Smart Grid, Semantic Interoperability, Real-Time Processing, OWL-Ontologies.

1. Introduction

According to the *National Institute of Standards and Technology (NIST)* (Von-Dollen (2009); NIST (2012)) and the *International Electrotechnical Commission (IEC)* (IEC (2010)), interoperability is one of the major challenges regarding smart grids. This fact is rooted in historical reasons: the components of the grid are highly heterogeneous in terms of vendors, data standards, communication protocols, and related software architectures that may be used, *e.g.* MOM (Message-Oriented Middleware), ESB (Enterprise Service Bus), SOA (Service-Oriented Architecture), CEP (Complex Event Processing), any of the Semantic Web components or SCADA (Supervisory Control And Data Acquisition) systems. Moreover, specific requirements of the electric business (such as the *real-time management* or the safety and criticality management) further complicate this scenario. Still, the most challenging requirement in a smart grid consists in the distribution of intelligence all over it. In other words, smart grids present many intelligent parts at different levels of their architecture (*e.g.* sensors, meters, substations, *etc*) that not only need to interchange data but also to contextualize and process it accordingly (ability known as *semantic interoperability*).

In this way, *semantic data models* are the basis that enable semantic interoperability within the grid. They consist in an abstract representation of a particular real domain, expressed through a specification language which may support knowledge inference, *e.g.* the description-logic-based languages DAML (Hendler and McGuinness (2000)), OIL (Bechhofer

et al. (2000)), OWL (Smith et al. (2004)), First-Order-Logic-based languages (Baral (2003)), etc. Achieving their full potential, however, requires considering the different technologies for semantic processing. In this way, there are a number of existing technologies that already support intelligence distribution in open distributed environments such as the one posed by the smart grid paradigm: multi-agent system platforms, Semantic Web technologies (specifically the combination of description logic reasoners, rule engines, and semantic repositories (Dustdar and Li (2011))), *etc.*

Another important issue is that semantic models for the Smart Grid domain need to be consumed in different formats, according to: 1) the specific technological requirements of the grid's systems, and 2) the requirements of the diverse knowledge abstraction levels. For instance, some systems may require semantic models provided as a Relational Database Model (RDBM), as an ontology for performing reasoning, or as Java classes. Similarly, knowledge abstraction levels involve supporting very specific domains of a smart grid, such as the electrical car or renewable energy. Their intelligent components do not need to know the overall knowledge model, but they do need to know the part related to their application domain. Here another concept arises that allows for the fragmenting of this general knowledge: the *semantic model's profile*. More accurately, a profile is a subset of the full semantic model according to its intended use in an application domain (IEC (2009)).

A naive approach for solving data interoperability is to write ad-hoc data interface programs for each pair of communicating systems. Nevertheless, experience shows that the development and maintenance of such programs is expensive both in terms of *time* and *effort* (King (2008)). In terms of *time* and effort, while it is very cheap and easy to develop a *traditional* or ad-hoc data interface against a semantic or generalized interface, experience shows that the maintenance of such programs will be more *expensive* (in the long term). Standardization has proven to be a much more effective strategy and a number of organizations are committed to the development of data-interchange format development according to specific application domains. In the context of electric models, the IEC has led one of the most important efforts in this way: the *Common Information Model (CIM)* defined by *IEC 61970* (IEC (2006, 2009)) and *IEC 61968* (IEC (2003 Ed.01)) defines a common vocabulary and a semantic model (officially provided as an UML model to be further exploited in several data models, *e.g.* OWL ontologies, relational data bases, RDF messages, *etc.*) for the main aspects related to the electric power industry, and it aims to allow application software to exchange

information about the configuration and status of an electrical network.

The IEC 61850 is another international standard developed by the EPRI and supported by the IEC. In particular, the IEC 61850 focuses on standardizing the design of electric substation automation. One of the most interesting features of the IEC 61850 is that it defines an abstract data model that can be mapped into different kinds of protocols.

Finally, the smart grid vision devises a number of real-time data sources, from the so-called smart meters to business data. This real-time data is typically delivered as streams of information (*e.g.* streams records), and, therefore, there is an additional need to provide more advanced solutions that deal with the *intelligent interpretation* of streams of data in real-time.

Against this background, we advance the state of the art in four main ways. First, we propose adopting an unique smart grid distributed knowledge base, based on the harmonization of two widely-established IEC industrial standards (*i.e.* CIM and the IEC 61850). Second, we put forward an architecture of distributed intelligent, active, and autonomous nodes spread all over the diverse smart grid application domains. Each node has a knowledge base *profile* (a portion of the global ontology and a set of axioms related to the application domain of the node), thereby achieving intelligence distribution and local decision making simultaneously. Third, we address the real-time requirements inherent to the smart grid by adding stream data processing tools (*i.e.* DDS as a real-time communication middleware and CEP as a pattern processing schema) to the architecture of the intelligent node. The combination with the local knowledge base profile allows an intelligent node to support semantic stream reasoning. Fourth, we close the gap existing between the smart grid *natural* domain and the world of agents. Namely, we integrate two main communication protocols: one based on IEC standards (CIM and the IEC 61850) and the other based on a standard multi-agent communication language (FIPA ACL), so every intelligent smart grid node will be able to communicate with both electrical devices and with multi-agent platforms such as JADE and JADEX.

The main benefit expected from this strategy relies on assuring and efficiently managing the interpretation of the interchanged data, considering the grid's requirements and design, thereby providing a step ahead to the established challenge for semantic interoperability in smart grids. A second and parallel result is that decision making capabilities will be supported along the smart grid systems through the intelligent data processing in each of the smart grid agents.

The remainder of the paper is structured as follows. Section 2 describes our general approach for leading with semantic interoperability, which is the first contribution of this paper. Section 3 presents a case study arising from the context of an ambitious research project in tight cooperation with a leading Spanish Utility. In particular, we illustrate the use of semantic data models to reach the vision of semantic interoperability in a particular smart grid architecture. Section 4 discusses the related work to our approach. Section 5 discusses the presented work, concludes and draws the avenues of future work. For readers who have no previous knowledge on UML models, Appendix A presents a small introduction to UML notation.

2. A Semantic Data Processing in Smart Grids

In this section, we will introduce our approach addressing *semantic interoperability* and *semantic model reasoning*. By a semantic model reasoning process, we understand a semantic reasoning process (*e.g.* description logic inferences, SWRL's inferences, *etc.*) that enables decision making in a smart grid. More accurately, we focus on the semantic processes involved in *real-time decision making*. We henceforth describe the general architecture of the smart grid (taking as reference model the ENERGOS smart grid (Penya et al. (2011))) in order to put in place the basic requirements that an intelligent node must fulfill.

2.1. Semantic Requirements of an Intelligent Node in a Smart Grid

The ENERGOS project aims at designing an intelligent network that can manage in real-time all multi-directional information flows that will appear in the new power network paradigm. This objective involves, for instance, the massive incorporation of renewable energy sources at different levels in the grid, greater participation of end-customers in their energy management, higher efficiency, and dealing with information and energy *prosumers* (individuals whose role may dynamically change from producer to consumer and vice versa) such as *electric vehicles* (that may additionally act as distributed batteries if required). The research comprises three main areas, as follows: *network infrastructure*, *data management platform*, and *intelligent management of network data*.

On a high-level view, the ENERGOS smart grid design presents a four-layered architecture, as depicted in Fig. 1. The keystone in the ENERGOS information processing system is the so-called PGDIN (*Power-Grid*

Distributed Intelligent Node), the intelligent node distributed across the architecture. Each layer simultaneously offers decision (according to the intended data use) and location abstraction (distribution or transport substations, Data Processing Center - DPC, or the entire grid). Level 0 comprises the *sensor and smart metering* layer (see marker **Level 0** in Fig. 1), in which the data is processed in real time (RT). Geographically, the data is captured and measured by the corresponding grid devices (e.g. I/O, sensors). Level 1 describes the *Data and Event Processing* layer (see marker **Level 1** in Fig. 1), in which the intelligent data processing and the Complex Event Processing (CEP) can present a combination between RT and quasi-RT, by the PGDINs; geographically, this stage is carried out at the secondary substation level. Level 2 represents the *Event Processing layer* (see marker **Level 2** in Fig. 1), in which CEP tasks are performed in quasi-RT by the PGDINs; geographically, this layer occurs at the secondary substation, primary substation and DPC levels. Finally, Level 3 identifies the *Business Management layer* (see marker **Level 3** in Fig. 1), in which PGDINs work with historical data; therefore, these business-oriented processes focus on the entire grid. All layers communicate with customer applications and the so-called *smart operation consoles*, through front-end applications.

Summarizing, the distribution of the PGDIN across the whole architecture enables the electrical information to be processed in an intelligent fashion, according to the localization and decision abstraction layers that are established in the ENERGOs smart grid architecture, and to the different smart grid time horizon requirements mentioned before. The functionalities implemented by a PGDIN depend on these requirements, and then, each PGDIN presents a specific configuration based on this core idea.

Taking into account that the ENERGOs smart grid is just a specific implementation of the smart grid paradigm, PGDIN are expected to interact not only with other PGDINs, but also with several intelligent components (such as the one introduced in (Ramchurn et al. (2011); Garcia et al. (2010))) in an ecosystem of heterogeneous smart grids models. This fact implies that each intelligent smart grid node has to be provided with semantic processing mechanisms to fulfill at least the following tasks:

- Support semantic data interchange between intelligent nodes in an open platform of smart grids.
- Enable semantic model reasoning to enable semantic decision making.

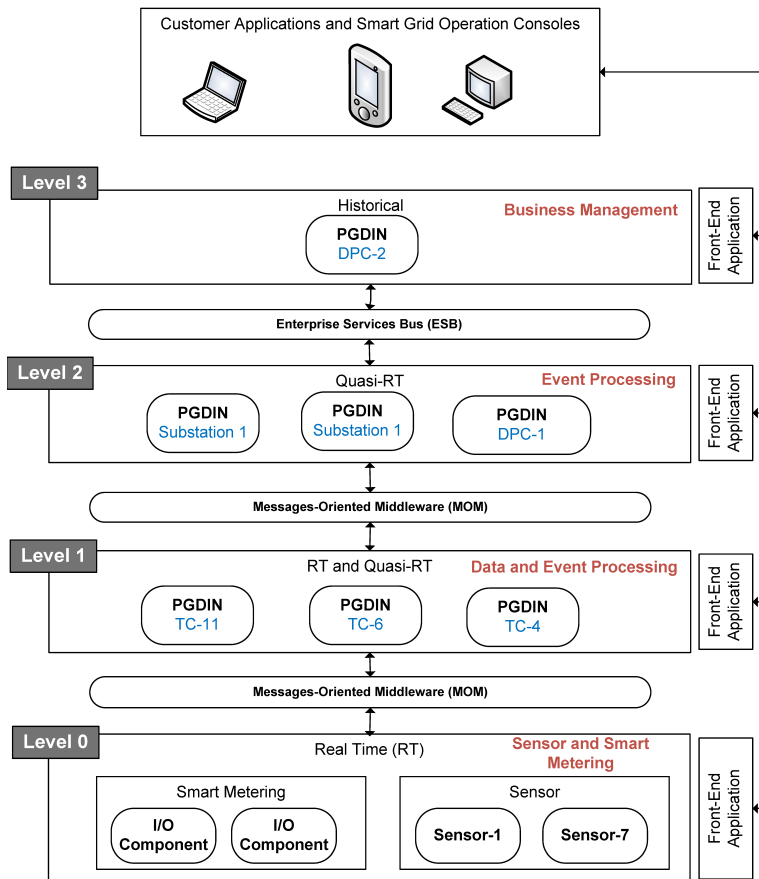


Figure 1: ENERGOS Real-Time Architecture in four layers.

These requirements emerge from the need to integrate network management, which is the real vision of a smart grid (Von-Dollen (2009); NIST (2012)). In the following subsection, we present a semantic approach to dealing with these semantic requirements of an intelligent node in a smart grid.

2.2. Intelligent Nodes in a Smart Grid: Grid Agents

In this section, we outline an approach for designing an open smart grid platform of intelligent, interacting nodes by considering basic concepts from the multi-agent system paradigm. Hence, by extending the basic idea of an intelligent agent (Wooldridge (1999)), we introduce the concept of *Grid-Agent*, an intelligent smart grid node which is able to:

- Manage an internal knowledge base. This knowledge base is designed according to the functional requirements of the node.
- Accomplish inference processes over *data streams* according to the time span and functional requirements of each node.
- Interchange data with other intelligent nodes or components that do not necessarily belong to the same smart grid.
- Be autonomous, proactive and reactive to the occurrence of unusual events.

2.2.1. First Step: Knowledge Base

The first step in the process of designing a grid agent is the definition of its internal knowledge base. The scope of the knowledge base of a grid agent is closely related to its functional requirements. Hence, this knowledge base is specific for each agent but its vocabulary must be understandable by the rest. This fact highlights the need for *standardized semantic data models* that define and establish a common language.

Any smart grid presents at least four separated data domains: *Assets*, *Communications*, *Connectivity* and *SCADA*. They are partially or fully covered by different industrial standards (*e.g.* IEC 61850, CIM, OAGIS, IEEE 1547, *etc.* ¹). Nevertheless, the NIST, EPRI, and IEC back the IEC 61850

¹The reader can find an exhaustive survey of industrial standards in the domain of Smart Grid in (Microsoft (2009))

and CIM as the *vertebral column* that may stand all data domains of any smart grid design. On the one hand, CIM addresses all aspects of assets, connectivity and SCADA systems; on the other, IEC 61850 deals with communications, connectivity and SCADA systems². There is an overlap between IEC 61850 and CIM in the data domains of Connectivity and SCADA systems, but this issue is currently being addressed by integrating their respective semantic data models (i.e. UML models and OWL ontologies) (Becker (2010); Hughes (2006)).

There is an official CIM semantic data model, currently maintained as an UML model³. It defines a common vocabulary and a *basic ontology* that can be transformed into an OWL ontology by tools such as CIMTool⁴. The IEC 61850 does not have an official semantic model yet; nevertheless, it is under construction by EPRI research groups using UML models Becker (2010); Hughes (2006).

Currently there are several approaches to constructing the UML models of IEC 61850 (Nieves et al. (2011)). Indeed, some of these approaches take the UML model of the CIM standard as their starting point. Hence, a unique semantic model between CIM and IEC 61850 can be designed by following the guidelines suggested by EPRIs reports (Becker (2010); Hughes (2006)). Therefore, let SG be a harmonized semantic data model between CIM and the IEC 61850. SG defines a basic vocabulary and a canonic basic ontology in the data domain of a smart grid, this ontology is denoted by O_{SG} . Hence, O_{SG} defines the *potential basic vocabulary* of any grid agent.

A knowledge base profile \mathcal{F} of a grid agent is a tuple of the form $\langle O, A \rangle$ in which:

- O is a subset of a harmonized ontology between CIM and IEC 61850, *e.g.* $O \subseteq O_{SG}$.
- A is a set of axioms *w.r.t.* O .

²It is worth mentioning that other standards and IEC working groups are now using the concepts of the IEC 61850 for their own domain; it includes the DER, the WindPower and the Hydro working groups; the standard is, therefore, gaining influence in many domains of the power utility automation.

³The reader can find the latest versions of CIM' semantic data model in <http://cimug.ucaiug.org/>

⁴<http://www.cimtool.org/>

Observe that O can be essentially regarded as a subontology of a harmonized ontology between CIM and the IEC 61850. For instance, this ontology can be constructed by tools such as the CIMtool. Moreover, A can be regarded as:

A semantic model: Given that the CIM XML data model is the same at the design and running time, a CIM XML data model can be understood as *a complete declarative specification* of a power electric network such that this declarative specification contains the complete description of the behavior of each power device that is present in a given grid.

A set of restrictions and inference rules: This means a set of restrictions over O (*e.g.* assumptions in the harmonization between CIM and IEC 61850) and a set of inference rules *w.r.t.* the vocabulary of O (*e.g.* SWRL rules (Horrocks et al. (2004))).

As a knowledge base profile defines an ontology, it allows us to perform inferences in order to support semantic decision making based on rules and continuous query/aggregation over RDF streams (meaning a sequence of triples continuously produced and annotated with timestamps from any sensor, device or asset that must be monitored or controlled). Semantic inference can be implemented in frameworks such as: SWRL (Horrocks et al. (2004)), SPARQL (Prud'hommeaux and Seaborne (2008)), Description Logic Inference (Hendler and McGuinness (2000); Bechhofer et al. (2000)), First-Order-Logic-based languages (Baral (2003)), etc. Fig. 2 depicts the exploitation and use of an instance of a knowledge base profile. The processes of continuous query/aggregation can be provided combining the mentioned frameworks with CEP engines and the capability they have to define event process windows over time.

2.2.2. Second Step: Message Communications

Once we have defined the way to construct a knowledge base of a grid agent, we move further to discuss how the data interchange between grid agents can be achieved. In this context, we want to point out that both CIM (IEC (2006)) and the IEC 61850 (IEC (2003)) already define XML message schemas. Hence, we assume that there is a mapping \mathcal{M} from a knowledge base profile into a particular CIM/IEC 61850 message, managed by each particular grid agent. Moreover, we assume that each grid agent

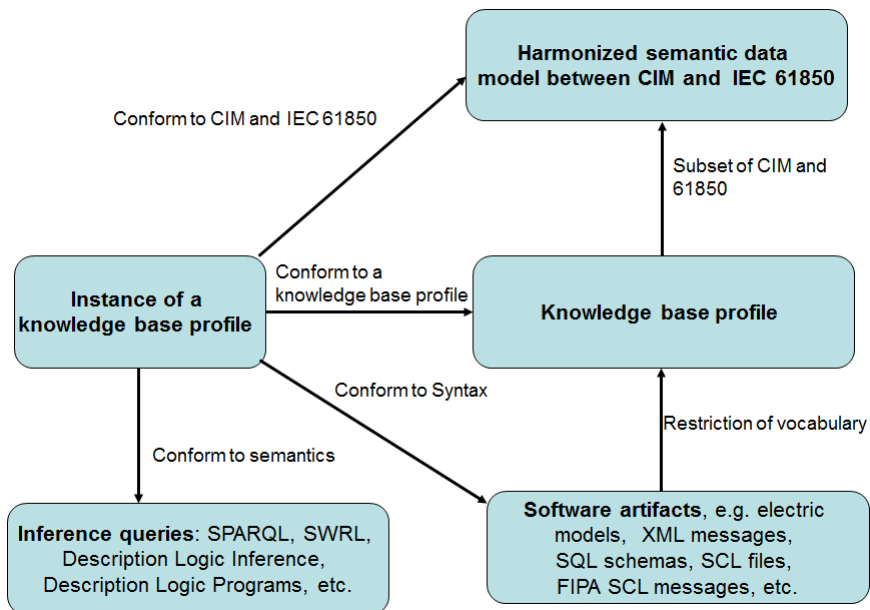


Figure 2: Exploitation of a knowledge base profile.

is FIPA-compliant and, therefore, understands and uses the FIPA Agent Communication Language (ACL) protocol⁵. Since in the FIPA ACL protocol it is assumed that each agent shares a common ontology, each FIPA agent intending to interact with a grid agent must own an ontology based on a harmonized ontology between CIM and IEC 61850 (this process is illustrated in Fig. 3). From this perspective, as previously mentioned, we open the possibilities of data interchange in an ecosystem of agent-based smart grids.

It is worth mentioning that the Message Transport Service (MTS) supports delivering FIPA ACL messages between agents on any given agent platform and between agents on different agent platforms. MTS can use different Message Transport Protocols (MTP) in order to handle the physical delivery of messages. Currently MTPs include HTTP, WAP and IIOP (Curry et al. (2003)). However, MTPs do not guaranty reliable messaging between the participants of a systems, *e.g.* FIPA agents. Enterprise-Messaging Services (EMS) or Message-Oriented Middelware (MOM) provide reliable messaging alternatives. For instance, DDS is a MOM which guaranties reliable messag-

⁵<http://www.fipa.org/repository/aclspecs.html>

ing. The integration of a MTS in a MOM requires to extend MTS in order to support actions of *subscription* and *publication* of messages in a MOM. In this setting, there are already extensions of MTS for supporting pub/sub schema over JMS (Curry et al. (2003)). Hence the interaction between systems which work under MTS and systems which work under a MOM as DDS is feasible, *e.g.* the interaction between grid agents and FIPA agents.

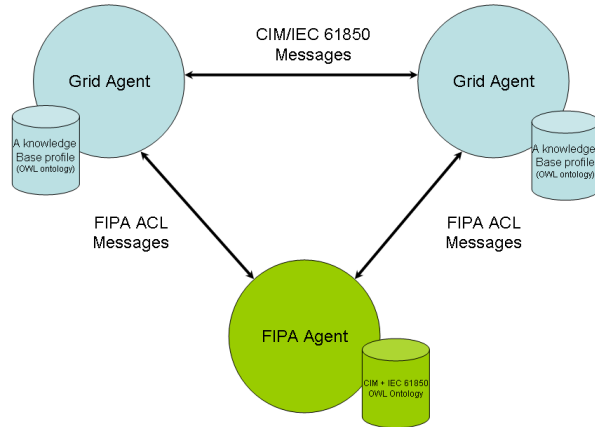


Figure 3: Message interchange between Grid Agents and FIPA Agents.

2.2.3. Third Step: BDI agents

Since it is expected that any intelligent node in a smart grid has to be autonomous, proactive and reactive upon unusual events, any grid agent presents an inference process based on *cognitive states of the world*. In particular, we assume that a grid agent is a BDI agent (Rao and Georgeff (1991)). It is worth mentioning that there are currently several platforms that support BDI agents (*e.g.* JADE⁶ or JADEx⁷).

3. A Case Study: Instantiating Grid Agents in PDGINS

In this section, we detail a specific implementation of grid agents, instantiated as Power-Grid Distributed Intelligent Nodes (PGDINs).

⁶<http://jade.tilab.com/>

⁷<http://jadex-agents.informatik.uni-hamburg.de/xwiki/bin/view/About/Overview>

As introduced in Section 2.1, the architecture of the ENERGOS smart grid is based on individual intelligent nodes, the PGDINs, spread all over the power network. Actually, PGDINs behave as grid agents with a number of potential aims and tasks: event processing, adaptive components, and business rules. Moreover, they support decision making upon data streams, and they can collaborate with other nodes in order to manage events and analyze situations that require it. Figure 4 portrays the general architecture of a PGDIN (see (Ortega et al. (2011)) for a more accurate account of the PGDIN). This architecture includes three main semantic components that enable semantic decision making in a PGDIN: a set of SWRL rules, an OWL-DL ontology and an OWL-DL/SWRL reasoner. Observe that the first two components characterize a knowledge base profile and the latter is an engine that supports inference queries *w.r.t.* the knowledge base profile.

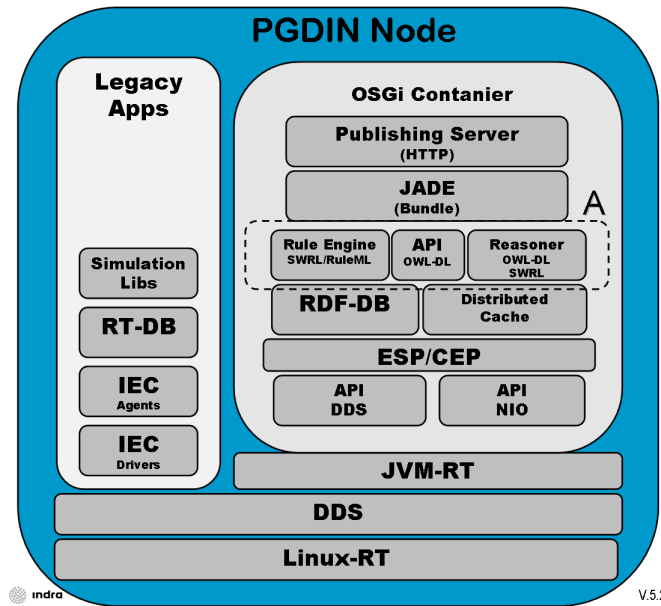


Figure 4: PGDIN node.

The process of decision making in a PGDIN is guided by two basic principles:

1. *Location Abstraction*: Each PGDIN is responsible for managing the relevant information on the set of assets that it controls (*e.g.* adding, removing, etc.). This means that an asset is only managed by one PGDIN.

2. *Decision Abstraction*: Each PGDIN is able to evaluate and take any decision that needs to be taken (*i.e.* business rules, calculation or inference) regarding the set of assets it manages.

These guidelines suggest that any PGDIN is autonomous enough to achieve auto-management. In this sense, the knowledge base profile of a PGDIN plays a central role for supporting semantic-based decision making and data streams analysis. Fig. 5 illustrates how the events (alarms, measurements, *etc.*) produced by various devices are grouped by a messaging middleware (DDS), then converted into Java objects, and, finally, injected into an event processing engine. This engine is responsible for detecting patterns of unusual conditions such as sudden changes in measured quantities, cutting conditions or supply problems.

Furthermore, the decision making in these situations often requires evaluating rules and semantically rich query execution. Additionally, the event processing engine is responsible for feeding information repositories based on two technologies:

1. A distributed cache of objects representing the state of the configuration and the real-time information on devices.
2. A semantic database storing information on events that must be addressed through collaboration with other agents (using the JADE framework) or semantic inference.

Now, we will introduce a real scenario in order to illustrate a semantic decision making process from a real time data stream in a PGDIN and the semantic interoperability between PGDINs.

3.1. Scenario

The first step in conducting this case study is the representation of the ENERGOS Smart Grid based on the CIM concepts. The semantic model has been specified in OWL 2 (OWL-Working-Group (2008)) and has been obtained from the 14th version of the CIM. The scenario consists of three electrical substations: two generator units and a consumer. In addition, all the items necessary to model the electrical network with the structure specified in CIM 14 have been created, such as electrical loads, AC line segments, voltage levels, current flow values, power transformers, *etc.* In Listing 1, a small part of a CIM-XML file is presented. This CIM-XML file

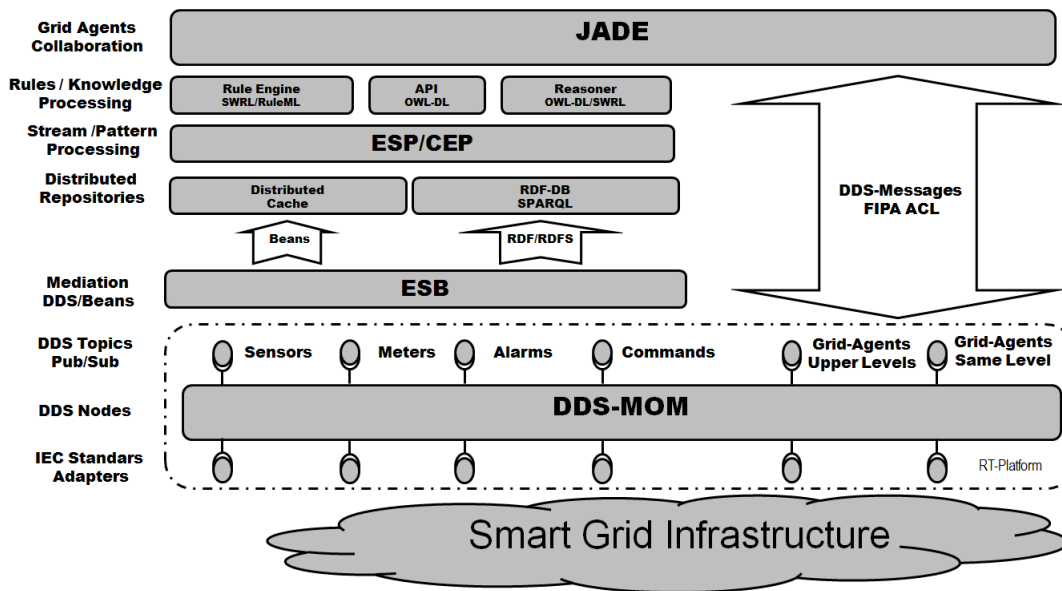


Figure 5: Semantic Data Interoperability in ENERGOS

illustrates the definition of three basic CIM 14 items: a GeographicalRegion, a Substation and Analog item.

```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cim="http://iec.ch/TC57/2008/CIM-schema-cim14#">
  <cim:GeographicalRegion rdf:ID="
    _37C0E10300D40CD812C47572C31C0AD">
    <cim:IdentifiedObject.name>GeoRegion</cim:IdentifiedObject
      .name>
    <cim:IdentifiedObject.localName>OpenCIM3bus</cim:IdentifiedObject
      .localName>
  </cim:GeographicalRegion>
<!--
-->
  <cim:Substation rdf:ID="_ID_SUBSTATION1">
    <cim:IdentifiedObject.name>Substation1</cim:IdentifiedObject
      .name>
    <cim:IdentifiedObject.localName>Substation1</cim:IdentifiedObject
      .localName>
    <cim:Substation.Region rdf:resource="#_
      _ID_SubGeographicalRegion"/>
  </cim:Substation>
  .
  .
  <cim:Analog rdf:ID="_D0E86D5181654BFDABCD81AD6F6A0E2F">
    <cim:Analog.positiveFlowIn>true</cim:Analog.positiveFlowIn<
      <
    <cim:Analog.normalValue>30.0</cim:Analog.normalValue>
    <cim:IdentifiedObject.name>Load_Q</cim:IdentifiedObject.
      name>
    <cim:Measurement.measurementType>ThreePhaseReactivePower<
      /cim:Measurement.measurementType>
    <cim:Measurement.MemberOf_PSR rdf:resource="#_
      _ID_SUB2_LOAD1"/>
    <cim:Measurement.Terminal rdf:resource="#_
      _LOADC5712A0D2F431594B0638457202CBC"/>
    <cim:Measurement.Unit rdf:resource="#_
      _96D78FFD1233469BA7683486BE983CD4"/>
  </cim:Analog>
  .
  .
</rdf:RDF>

```

Listing 1: An example of a CIM-XML file. A CIM-XML file defines an electric model which is the same at the design and running time.

CIM 14 states that each resource of the power system needs to be connected to the network through a terminal, which is in turn associated with a connectivity node, which is the object that interacts directly with the grid. Fig. 6 shows the representation of a power grid following the CIM notation.

The consumer substation (Substation 2 in Fig. 6) has been associated to a real Spanish substation named SALAS. The SALAS substation is managed by the PGDIN3 (an instance of the PGDIN architecture previously

presented), which is part of the ENERGOs implemented solution. The aim is to use the collected data of the real SALAS substation to perform usual validation processes of the simulation and operation systems, by following our approach. For this purpose, we have gathered data of real value curves of the SALAS substation for: *active power, reactive power, voltage and current*. The measures were taken every 15 minutes (period corresponding to the Spanish current policy in its grid meter systems⁸). These data are currently used to perform the simulation of electric charges, to check their consistency and to issue warnings to the grid operator in case deviations are detected.

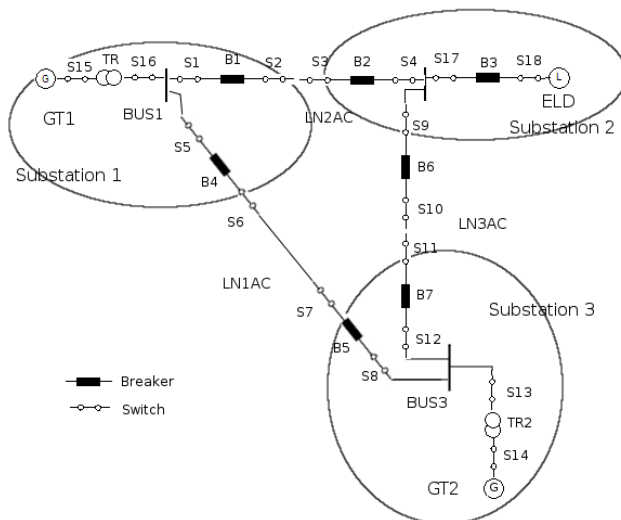


Figure 6: A CIM model for a three sub-stations system (Shahidehpour and Wang (2003)).

Before moving on, we want to point out that since a CIM-XML file defines an electric model that is the same at the design and running time, a CIM XML data model of the CIM model presented in Fig. 6 defines a *complete declarative specification* of the power electric network. Hence, a CIM XML data model will be part of the knowledge base of each PGDIN in the power network. More accurately, each PGDIN will have the CIM XML data model that defines the assets that it controls. In this setting, each PDGIN will be

⁸These values have been provided by the Spanish Gas Natural/Union Fenosa utility company.

able to perform semantic model reasoning by considering a CIM XML data model. In the following sections, we will illustrate how a CIM XML data model can be exploited for semantic processing.

3.2. Business Logic Rule

Now that we have defined the structure of our power network, we are going to introduce some business logic rules, which have to be considered in the interaction between PGDINs. Specifically, our case study focuses on the simulation and operation systems. Typical validations of these systems include consistency checking over the data registered by value curves of: *active power, reactive power, voltage and current*. Furthermore, they are in charge of providing the technical operators with the knowledge for helping them to diagnose problems and identify solutions. These systems meet their objectives by accessing the grid information through the different grid nodes.

A typical business rule is triggered when the measures are required to perform the consistency checking over the curves. Then the process starts with the Energy Management System (EMS), which requires the 15' measures of a specific day from the SALAS substation. This command is sent as an event to the messaging bus (DDS) so that the corresponding PGDIN can calculate and send back the requested measures (in Fig. 7 the StartEvent point). In this case, it is PGDIN3 (controlling the SALAS substation) which identifies through the messaging bus that it needs to retrieve the 15' measures for that specific day. The required data processing workflow is initiated by PGDIN3 at the Business Management Level (BPEL⁹ process in Fig. 7). If PGDIN3 has the complete measures in its local knowledge base, then it executes its own business services (by the SOA¹⁰ bus) to retrieve the data and send them back to the inquiring agent (i.e. the EMS in this case, see Fig. 7, the first decision point). If PGDIN3 does not have the complete measures, it triggers an event to the messaging bus to inquire for a provider of the missing measures. Eventually, PGDIN3 reads a message from a PGDIN provider with the encoding information to invoke that provider PGDIN's services (in Fig. 7, the BPEL process called "Affected PGDIN identifies a message from a Provider"). PGDIN3 invokes the provider PGDIN's services and receives the required measures. PGDIN3 checks that the measures

⁹Business Process Execution Language.

¹⁰Service Oriented Architecture.

are complete (in Fig. 7, the second decision point). If they are, PGDIN3 sends back the required measures to the inquiring agent (i.e. the EMS) and the BPEL process ends at this point. If the measures are not complete, PGDIN3 triggers again an event asking for the missing measures until the complete measures are ready. Once the measures are complete, they are sent back to the inquiring agent (the EMS) and the BPEL process ends at this point.

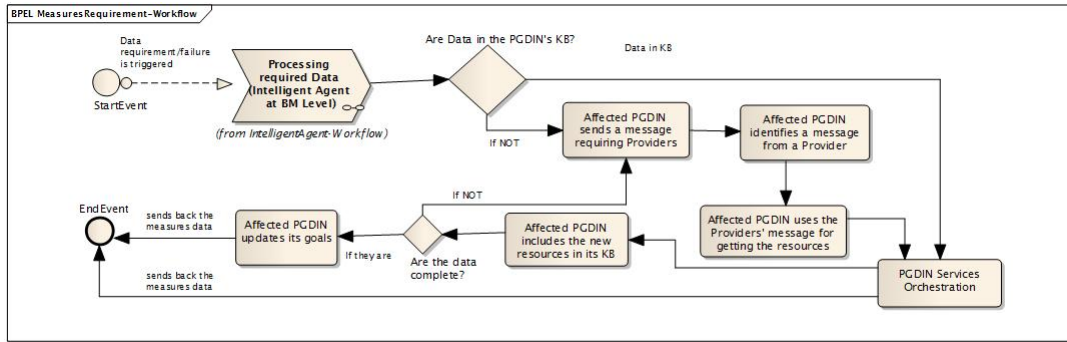


Figure 7: BPEL Work flow: Measures Requirements

3.3. Performing Semantic Processing

This section focuses on the internal semantic model reasoning process of a PGDIN. In the previous section, the business rule for collecting the measures curves that are required for an agent as the EMS involved verifying if the corresponding PGDIN in charge had these measures in its knowledge base. In a negative case, the measures should be retrieved from other provider PGDINs. In a case in which the PGDIN has the measures, then the corresponding semantic processing is triggered. The technical components needed to this end in each PGDIN are:

1. **Pellet**¹¹: This reasoner offers full integration with open source and commercial semantic repositories and with ontology development tools;
2. **Jess**¹²: Jess has been selected as the rule engine due to its stability and development, marketing and research support offered;
3. **Jena**¹³: This API offers integration with Pellet and supports manipu-

¹¹<http://clarkparsia.com/pellet/>

¹²<http://www.jessrules.com>

¹³<http://jena.sourceforge.net/>

lation of OWL models. These features make Jena one of the best APIs for managing the semantic CIM model and the knowledge base of this approach, through Java-based systems.

Currently, simulation and operation systems work with historical data to perform validations in specific timestamps or taking the values of a complete day. Typical validations to identify deviations in the value curves are the following:

1. Check whether *Power Module Exceeds Fixed Limits*, where P is active power and Q reactive power. For instance:

$$if(G = (\sqrt{P^2 + Q^2})) > 20$$

2. Calculate the *Power Module* from current and voltage, where current is Int and voltage is T :

$$I = (\sqrt{3} * T * Int/1000)$$

3. Check the consistency of P , Q , Int , T :

$$if(|G - I| = 0, 0, |G - I|/|G| < 0, 2$$

These validation rules have been introduced into the knowledge base using Protégé. For illustration purposes, the same environment is used to show the rule execution and the resulted predicates that are injected back into this knowledge base and, therefore, we have used the Jess integration with Protégé. Table 1 illustrates the SWRL rule *Power Module NOT Exceeding Fixed Limits* for the substation SALAS. The results of this rule execution give all measures of P , Q that have a *Power Module* that does not exceed a particular fixed limit (20 in this example). Let us observe that the design of the SWRL rules such as the one presented in Table 1 is mainly based on the semantic model of the CIM concepts. This means that by considering an OWL ontology of the CIM semantics model and an electric model in CIM-XML format, we have a complete knowledge base for performing semantic reasoning.

Table 2 shows the results of the SWRL rule of Table 1 for only 10 measures of P , Q . Thereafter, all previous validations are also introduced as SWRL rules into the three-substation model following the same procedure. Since this semantic model (CIM) is part of the PGDIN node of the ENERGOS

Power Module do NOT exceeds fixed limits

```
cim:AnalogValue(?pact) ∧ cim:AnalogValue.Analog(?pact,?Pa) ∧
cim:IdentifiedObject.name(?Pa, "Load_P") ∧
cim:MeasurementValue.timeStamp(?pact,?tact) ∧
cim:AnalogValue(?prea) ∧ cim:MeasurementValue.timeStamp(?prea,?trea) ∧
swrlb:equal(?tact,?trea) ∧ cim:AnalogValue.Analog(?prea,?Pr) ∧
cim:IdentifiedObject.name(?Pr, "Load_Q") ∧
cim:AnalogValue.value(?pact,?pactiva) ∧
cim:AnalogValue.value(?prea,?preactiva) ∧ swrlb:pow(?PA,?pactiva,2)
∧swrlb:pow(?PR,?preactiva,2)∧
swrlb:add(?sum,?PA,?PR) ∧ swrlb:pow(?G,?sum,0.5) ∧
swrlb:lessThan(?G,20) →
sqwrl:select(?tact,?pactiva,?preactiva,?G) ∧ sqwrl:orderBy(?tact)
```

Table 1: This validation has been introduced into the three-substation model as a SWRL rule. The clause *select* at the end allows us to visualize measures that have passed the validation.

smart grid architecture, it will then be possible to execute validations at all architecture levels: sensor and smart metering, data and event processing, event processing, and business management layers. Moreover, the validation can be performed in RT, Quasi-RT and H.

The introduction of the validations as SWRL rules into the three-substation model provides the smart grid systems with at least two advantages:

1. Consistent information, since the reasoner (in this case Pellet) checks the consistency of the semantic model, according to the smart grid time slots.
2. To infer new information, since the rule engine (in this case Jess) executes the validation as SWRL rules, obtaining not only the calculations typically performed with a spreadsheet as occurs nowadays, but also inferred information related to the identification of the grid's deviations.

The need for updating the knowledge base profile of a PGDIN can emerge from different situations. For instance, if a new assert is installed. In this scenario, the electric model of the electric power network will indeed change. This electric model can be exported into a CIM XML file. Since CIM models

Table 2: The table shows ten (P, Q) measures with a period of 15", and the *Power Module* that has not exceeded a specific fixed limit (20 in the SWRL rule of Table 1).

Measure	Time-stamp	Active Power MW (P)	Reactive Power MVAR (Q)	Power Module MVA for (P,Q)
SALAS_0001	2010-07-05 T00:00:00	2.41	1.45	2.812578888
SALAS_0002	2010-07-05 T00:15:00	2.41	1.45	2.812578888
SALAS_0003	2010-07-05 T00:30:00	2.41	1.45	2.812578888
SALAS_0004	2010-07-05 T00:45:00	2.41	1.45	2.812578888
SALAS_0005	2010-07-05 T01:00:00	5.79	1.45	5.968802225
SALAS_0006	2010-07-05 T01:15:00	5.79	1.45	5.968802225
SALAS_0007	2010-07-05 T01:30:00	5.79	1.45	5.968802225
SALAS_0008	2010-07-05 T01:45:00	5.79	1.45	5.968802225
SALAS_0009	2010-07-05 T02:00:00	5.79	1.45	5.968802225
SALAS_0010	2010-07-05 T02:15:00	5.79	-0.08	5.790552651

are the same at design and running time, the PGDIN in charge of the new asset will only have to update its CIM-XML file containing the new asset. This process can be regarded as an update of the knowledge base file of the PGDIN. By observing the architecture of a PDGIN (see Figure 4), we can see that a PDGIN is basically an OSGi¹⁴ container whose components can be managed by the so called OSGi bundles. One of these OSGi bundles is a set of the semantic rules (SWRL rules) in a XML file. In this setting, we can use different approaches for updating OSGi bundles (which means to update the SWRL rules of a PDGIN). For instance, given that OSGi supports FTP-services, the set of SWRL rules (a OSGi bundle) of a PDGIN can be updated by FTP services. Indeed, the update of the semantics rules can be done on the fly (*i.e.* in real-time) following different approaches as the one presented in Shen et al. (2010) .

3.4. Extension of the Semantic Data Model

So far, we have shown that the knowledge base of each PGDIN has to be based on the vocabulary of a potential ontology, which can emerge from the harmonization between CIM and ICE 61850. It is clear, however, that this harmonized ontology will not contain concepts from emerging data domains such as *electric vehicles*. Actually, this harmonized ontology between CIM and ICE 61850 is usually regarded as the *canonic ontology* in this domain (*i.e.* the starting ontology that can be extended to involve other domains). Hence, in this section, we introduce a hypothetic extension of the CIM semantic data model which allows us to capture concepts from the electric vehicle domain. In particular, we introduce new hypothetical concepts to our canonic ontology to deal with the management of recharge points for electric vehicles, as depicted in Fig. 8. For readers who have no previous knowledge in UML notation, in Appendix A a small introduction to UML notation is presented.

Let us observe in Fig. 8 that the new classes are subclasses from the `IdentifiedObject` class, which is a class from the `CORE` package from CIM. On the other hand, new classes use existing CIM classes to define novel data properties.

3.5. Semantic interoperability

Finally in this subsection, we will illustrate the role of the semantic data model in each PGDIN for performing semantic interoperability.

¹⁴<http://www.osgi.org>

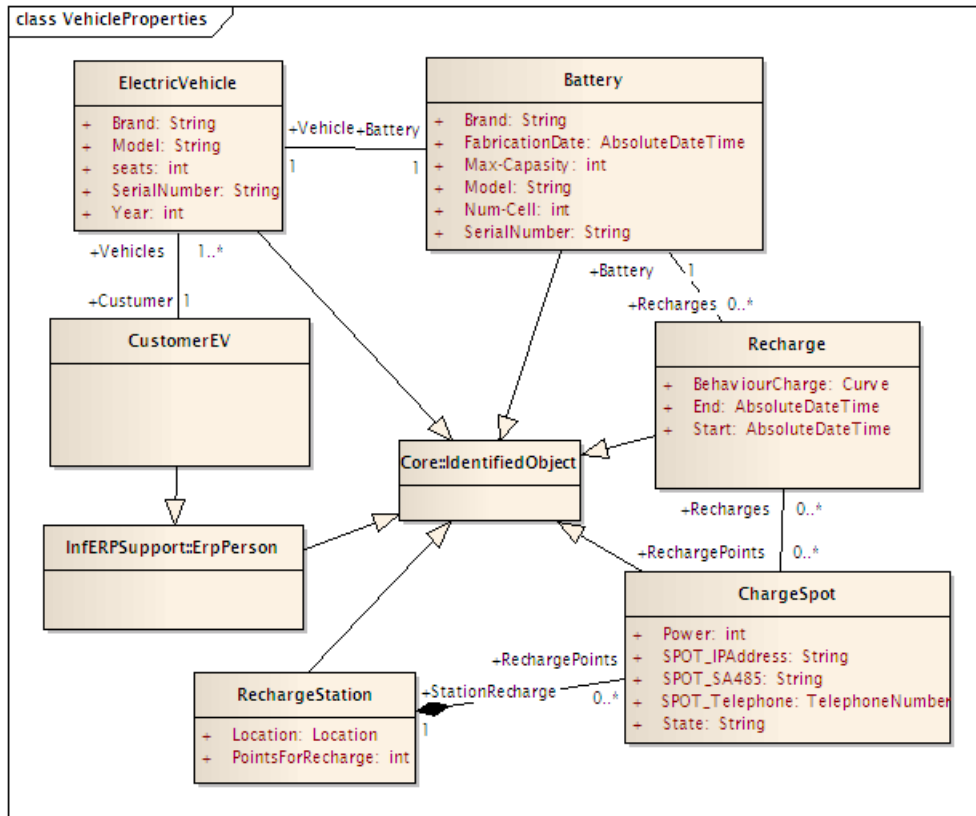


Figure 8: An hypothetical extension of the CIM semantic model to deal with the management of recharge points for electric vehicles.

Let $\langle O_A, SWRL_A \rangle$ be a knowledge base profile of a $PGDIN_A$. O_A is an OWL ontology, which is a subset of the CIM ontology and $SWRL_A$ defines a set of inference rules restricted to the vocabulary of O_A . By using O_A , two PGDINs can interchange electric models such as the presented in Listing 1.

Observe that in Listing 1, the CIM-XML file has a header defining the name space in which the items of the electric model are defined. More accurately, O_A is a subset of the name space:

`http://iec.ch/TC57/2008/CIM-schema-cim14#$`

Since O_A is a subset of the vocabulary of this name space, two PGDINs could interchange measurement values of substations by delivering CIM-XML files such as the one presented in Listing 2. We can observe that the CIM-XML file presented in Listing 2 is defining a set of measurement values of an analog device which is identified as: “#_D0E86D5181654BFDABCD81AD6F6A0E2F”. This device is part of a power network. In particular, this device was defined in Listing 1. Hence, this means that the measurement values which appear in the CIM-XML file of Listing 2 can be interpreted by using the electric model of Listing 1. Moreover we can see that the RDF files, as the CIM-XML file of Listing 2, define instances of the classes contained in the PGDIN OWL ontology (*e.g.* O_A).

```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cim="http://iec.ch/TC57/2008/CIM-schema-cim14#">
<cim:AnalogValue rdf:ID="_0001R">
  <cim:AnalogValue.value rdf:datatype="http://www.w3.org/2001/XMLSchema#float"> 1.45 </cim:AnalogValue.value>
  <cim:MeasurementValue.timeStamp rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime"> 2010-07-05T00:00:00 </cim:MeasurementValue.timeStamp>
  <cim:AnalogValue.Analog rdf:resource="#D0E86D5181654BFDABCD81AD6F6A0E2F" />
</cim:AnalogValue>
<cim:AnalogValue rdf:ID="_0002R">
  <cim:AnalogValue.value rdf:datatype="http://www.w3.org/2001/XMLSchema#float"> 1.45 </cim:AnalogValue.value>
  <cim:MeasurementValue.timeStamp rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime"> 2010-07-05T00:15:00 </cim:MeasurementValue.timeStamp>
  <cim:AnalogValue.Analog rdf:resource="#D0E86D5181654BFDABCD81AD6F6A0E2F" />
</cim:AnalogValue>
.
.
.
</rdf:RDF>

```

Listing 2: A CIM-XML file which characterizes a set of measurements from the substation SALAS.

As previously shown in Fig. 5, each RDF instance is converted into Java beans and managed by the PGDIN’s distributed cache.

Finally, we can observe that the knowledge base profile of any PGDIN presents two main objectives:

1. Since the ontology of any knowledge base profile is a subset of the harmonized semantic data model between CIM and IEC 61850, this ontology defines a proper vocabulary for interchanging data in a smart grid domain.
2. Since a knowledge base profile is a particular vocabulary of each PGDIN, this knowledge base supports the internal semantic-based decision making processes of any PGDIN.

Before finishing this section, it is worth mentioning that the design of each profile is unique with respect to each PDGIN. In particular, each profile is designed according to the set of assets that will be managed for a given PDGIN. Hence the profile of each PDGIN is unique. Therefore, the ontology of a profile gives a description of each asset that has to be managed for a

given PDGIN. This means that if we apply a joining of all profiles, the merged ontology will only contain the set of assets which belong to the whole smart grid. On the other hand, any asset is only managed by one PDGIN. This means that a particular instance of a class (which belongs to a particular ontology) will be unique (this suggests that the `identifiedobject` is unique in the whole smart grid). Hence, the intersection between the sets of instances of CIM-classes which is managed by different PDGINs will be empty. Indeed, an instance of a class with an `identifiedobject` X is only managed by one PDGIN.

4. Related Work

In response to the challenges posed by the smart grid vision, several intelligence platforms have been introduced; for instance, (Catterson et al. (2011)) discussing the processing of electric grid data through a number of artificial intelligence techniques (*e.g.* constraint programming or rule-based expert systems considering fuzziness) applied to diverse application domains such as autonomous control of distribution networks, condition monitoring, post-fault analysis or voltage sag and swell monitoring. To achieve this goal, they also put forward a multi-agent architecture but do not go beyond stating that CIM and the IEC 61850 should be the key standard for syntactical interoperability (not semantic). (Almeida and Kagan (2011)) focus only on the hardware and software architecture required to support this data processing in different time spans (real time, quasi real-time and historical). (Garcia et al. (2010)) address similar challenges to ours but their work is too focused on multi-agent system aspects (*e.g.* all agents interactions exclusively performed in FIPA ACL) and do not clarify whether they support real-time stream processing. Moreover, they introduce agents having a knowledge base expressed in rules but do not specify which kind of inference machine they use. (Ramchurn et al. (2011)) present an approach based on a FIPA-compliant multi-agent system to reduce household energy bills, but their focus is smaller than ours, since we address the whole smart grid. In Cohen (2008), a multi-agent platform has been introduced. According to the author, this platform is based on the so called GridAgentTM Framework. This agent framework allows us to define different kinds of agents in order to support the management of distributed energy resources. However, it seems that these agents do not allow semantic reasoning, which is one of the main features of a PDGIN.

As has been stressed in Section 2.2.1, given that the CIM XML data model is the same at the design and running time, a CIM XML data model can be understood as *a complete declarative specification* of an electric power network such that this declarative specification contains the complete description of the behavior of each power device that is present in a given grid. In this setting, the approaches which are based on *model-based reasoning* are quite similar to the approach presented in this paper (Davidson et al. (2003, 2006)). A nice feature of our approach is that the semantic model of an electric power network are understandable by different platforms which are CIM compliant.

5. Conclusions

Semantic technology have been successfully applied to solve several interoperability problems in open platforms such as distributed systems. We have address in this paper the application of semantic and syntactic interoperability to the smart grid vision. Further, we believe that semantic technology defines a solid frame to support semantic decision making

According to the EPRI and the NIST, interoperability in electric distributed management systems is one of the main challenges to be tackled in order to achieve the envisioned smart grid. On the other hand, another big challenge is providing the tools to allow the smart grid to manage itself. In order to face these challenges, each node in a smart grid must possess both reasoning ability and a knowledge base upon which the decision making will take place. In this setting, in Section 2, we have introduced the concept of *grid agent* to enable this scenario. The internal knowledge of each grid agent is condensed on *a knowledge base profile*, combining an OWL ontology and a set of axioms (either a set of restriction of the ontology or a set of inference rules). Moreover, the ontology of a knowledge base profile is built as a subset of the harmonized semantic data model of two IEC standards: the CIM and the IEC 61850. Since they define a common vocabulary in the data domain of a smart grid, each grid agent is able to interchange information with any electric device that understands either CIM or IEC 61850. Moreover, we assume that each grid agent is FIPA-compliant and, therefore, speaks FIPA ACL; this feature will enable grid agents to communicate with any FIPA ACL multi-agent platform. Furthermore, each grid agent may perform decision making by using its knowledge base (inference rules), defined according to the vocabulary of its ontology. It is worth mentioning that although the

inference can be achieved by rules such as SWRL rules or rules restricted to the inferences of description logic, the grid agent can additionally be provided with alternative inference engines such as description logic programs (Grosz et al. (2003)) or HEX-programs (Eiter et al. (2010)), in order to manage different kinds of inferences.

In Section 3, we have instantiated the idea of a grid agent into a particular implementation of an intelligent node, the so-called PGDIN. The PGDIN architecture presents a knowledge base profile and, additionally, tools to enable stream data processing (*e.g.* DDS and CEP). This combination allows an intelligent node to support semantic stream reasoning in order to deal with decision making in real-time. Further, it enables a PGDIN to support semantic stream reasoning similar to C-SPARQL (Barbieri et al. (2010)). Possibly one of the main weaknesses of using a knowledge base profile in real-time processes lies in the fact that nowadays the majority of the reasoner engines are very time consuming. Nevertheless, since the vocabulary of a knowledge base profile is restricted to the functional requirements of each grid agent, the real-time reaction of each intelligent node is closely related to its functional requirements. This aspect suggests that each grid agent can be designed for different real-time requirements.

Each knowledge base profile of a grid agent is based on an unique smart grid distributed knowledge base; however, with the incorporation of, for instance, novel renewable energy sources, this knowledge base has to be updated continuously. This need could be regarded as a weakness of the approach; still, since the presented approach is based on ontology, the process of updating a knowledge base of a grid agent is solely restricted to the updating of its knowledge base profile. For instance, by extending a knowledge base profile by the extension of the CIM semantic model introduced in Section 3.4, a PDGIN will be able to understand concepts and support decision making in terms of recharge points for electric vehicles.

Future work will focus on extending the harmonized semantic data model of CIM and IEC 61850 in order to capture domains such as wind energy. Moreover, we are currently working on the construction of a proof-case prototype in which different PGDINs interact in real time. This proof-case requires implementing a DDS middleware upon which an ecosystem of PGDINs interact.

Acknowledgment

This work has been supported by the ENERGOS Project funded by the Spanish MEC (ENERGOS TIN2009-00000). The authors would like to acknowledge the contributions of their colleagues from the ENERGOS consortium. The views expressed in this paper are not necessarily those of the ENERGOS consortium. Angelina Espinoza started developing part of the research presented in this paper while she was working for the SYST group at the Technical University of Madrid.

References

- Almeida, C. F. M., Kagan, N., 2011. Using genetic algorithms and fuzzy programming to monitor voltage sags and swells. *IEEE Intelligent Systems* 26 (2), 46–53.
- Baral, C., 2003. *Knowledge Representation, Reasoning and Declarative Problem Solving*. Cambridge University Press, Cambridge.
- Barbieri, D., Braga, D., Ceri, S., Grossniklaus, M., 2010. An Execution Environment for C-SPARQL Queries. In: *Proc. Intl. Conf. on Extending Database Technology, EDBT*. pp. 441–452.
- Bechhofer, S., Broekstra, J., Decker, S., Erdmann, M., Fensel, D., Goble, C., van Harmelen, F., Horrocks, I., Klein, M., McGuinness, D., Motta, E., Patel-Schneider, P., Staab, S., Studer, R., 2000. An informal description of Standard Oil and Instance OIL. Tech. rep.
URL <http://www.ontoknowledge.org/oil/down1/oil-whitepaper.pdf>
- Becker, D., May 2010. Harmonizing the International Electrotechnical Commission Common Information Model (CIM) and 61850. Tech. Rep. 1020098, Electric Power Research Institute (EPRI).
- Catterson, V. M., Davidson, E. M., McArthur, S. D., 2011. Embedded intelligence for electrical network operation and control. *IEEE Intelligent Systems* 26 (2), 38–45.
- Cohen, D. A., 2008. Gridagents: Intelligent agent applications for integration of distributed energy resources within distribution systems. In: *IEEE*

- Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century. IEEE Press, pp. 1–5.
- Curry, E., Chambers, D., Lyons, G., 2003. A JMS Message Transport Protocol for the JADE Platform. *Intelligent Agent Technology, IEEE / WIC / ACM International Conference on* 0, 596.
- Davidson, E. M., McArtur, S. D., McDonald, J. R., 2003. A toolset for applying model-based reasoning techniques to diagnostics for power systems protection. *IEEE Transactions on Power Systems* 18 (2), 680–687.
- Davidson, E. M., McArtur, S. D., McDonald, J. R., Cumming, T., Watt, I., 2006. Applying multi-agent system technology in practice: automated management and analysis of scada and digital fault recorder data. *IEEE Transactions on Power Systems* 21 (2), 559–567.
- Dustdar, S., Li, F. (Eds.), 2011. *Service Engineering - European Research Results*. Springer.
- Eiter, T., Ianni, G., Schindlauer, R., Tompits, H., 2010. Towards Efficient Evaluation of HEX-Programs. In: Hunter, J. D. . A. (Ed.), *Proceedings of the 11th Workshop on Nonmonotonic Reasoning*. pp. 40–46.
- Garcia, A. P., Oliver, J., Gosch, D., 2010. An Intelligent Agent-Based Distributed Architecture for Smart-Grid Integrated Network Management. In: *1st IEEE Workshop on Smart Grid Networking Infrastructure*. Denver, Colorado, pp. 1013–1018.
- Grosf, B. N., Horrocks, I., Volz, R., Decker, S., 2003. Description logic programs: combining logic programs with description logic. In: *Proceedings of the 12th international conference on World Wide Web. WWW '03*. ACM, New York, NY, USA, pp. 48–57.
URL <http://doi.acm.org/10.1145/775152.775160>
- Hendler, J., McGuinness, D. L., November/December 2000. The DARPA Agent Markup Language. *IEEE Intelligent Systems Trends and Controversies* 15 (6), 67–73.
- Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosf, B., Dean, M., 2004. SWRL: A Semantic Web Rule Language Combining OWL and

RuleML.

URL <http://www.w3.org/Submission/SWRL/>

Hughes, J., December 10 2006. Harmonization of IEC 61970, 61968, and 61850 Models. Tech. Rep. 1012393, Electric Power Research Institute (EPRI).

IEC, 2003. Communication networks and systems in substations - introduction and overview. Tech. rep., International Electrotechnical Commission.

IEC, 2003 Ed.01. IEC 61968 application integration at electric utilities - system interfaces for distribution management - part 11: Common Information Model (CIM).

IEC, 2006. IEC 61970-501: Energy management system application program interface (EMS-API) Part 501: CIM RDF Schema.

IEC, 2009. IEC 61968-9 Application integration at electric utilities - System interfaces for distribution management, Part 9: Interfaces for meter reading and control. IEC - International Electrotechnical Commission, 1st Edition.

IEC, 2009. IEC 61970-301: Energy management system application program interface (EMS-API) Part 301: Common Information Model (CIM) Base.

IEC, June 2010. IEC Smart Grid Standardization Roadmap.
URL www.iec.ch/smartgrid/downloads/sg3_roadmap.pdf

King, L., November 2008. The common information model for distribution - an introduction to the CIM for integrating distribution applications and systems. Tech. Rep. 1016058, Electric Power Research Institute (EPRI).

Microsoft, October 2009. Microsoft Power and Utilities - Smart Energy Reference Architecture (SERA) . www.Microsoft.com/Utilities.

Nieves, J. C., de Mues, M. O., Espinoza, A., Rodríguez-Álvarez, D., October 2011. Harmonization of semantic data models of electric data standards. In: 9th IEEE International Conference on Industrial Informatics - ICT @ SmartGrids. IEEE Press, pp. 733–738.

- NIST, February 2012. NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 2.0. NIST Special Publication 1108R2, NIST - National Institute of Standards and Technology.
 URL http://www.nist.gov/smartgrid/upload/NIST_Framework_Release_2-0_corr.pdf
- OMG, 2011. OMG Unified Modeling Language™ (OMG UML), Infrastructure. Deliverable, Object Management Group.
- Ortega, M., Alvarez, A., Espinoza, A., Garbajosa, J., 2011. Towards a Distributed Intelligence ICT Architecture for the Smart Grid. In: Proceedings of the IEEE 9th International Conference on Industrial Informatics. IEEE Press, Caparica Lisbon, Portugal, pp. 745–749.
- OWL-Working-Group, W., 2008. OWL 2 web ontology language.
 URL <http://www.w3.org/TR/owl2-overview/>
- Penya, Y. K., Garbajosa, J., Ortega, M., González, E., 2011. ENERGOS: Integral smart grid management. In: Proceedings of the IEEE 9th International Conference on Industrial Informatics. IEEE Press, pp. 727–732.
- Prud'hommeaux, E., Seaborne, A., 2008. SPARQL Query Language for RDF.
 URL <http://www.w3.org/TR/rdf-sparql-query/>
- Ramchurn, S. D., Vytelingum, P., Rogers, A., Jennings, N. R., July 2011. Agent-based homeostatic control for green energy in the smart grid. ACM Transactions on Intelligent Systems and Technology 2 (4), 1–28.
- Rao, A. S., Georgeff, M. P., 1991. Modeling Rational Agents within a BDI-Architecture. In: Allen, J., Fikes, R., Sandewall, E. (Eds.), Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning. pp. 473–484.
- Shahidehpour, M., Wang, Y., 2003. Communication and Control in Electric Power Systems - Applications of Parallel and Distributed Processing. IEE Press Power Engineering Series.
- Shen, F., Du, S., Huang, L., 2010. A dynamic update framework for osgi applications. In: Zhang, W., Chen, Z., Douglas, C., Tong, W. (Eds.), High Performance Computing and Applications. Vol. 5938 of Lecture Notes in

Computer Science. Springer Berlin Heidelberg, pp. 350–355.
URL http://dx.doi.org/10.1007/978-3-642-11842-5_48

Smith, M. K., Welty, C., McGuinness, D. L., 2004. OWL Web Ontology Language Guide.
URL <http://www.w3.org/TR/owl-guide/>

Sparx-Systems, 2012. Uml 2 tutorial. Tech. rep.
URL http://www.sparxsystems.com.au/resources/uml2_tutorial/

Von-Dollen, D., 2009. Report to NIST on the Smart Grid Interoperability Standards Roadmap - Post Comment Period Version. Deliverable, Electric Power Research Institute (EPRI).

Wooldridge, M., 1999. Multiagent Systems A Modern Approach to Distributed Artificial Intelligence. The MIT press, Ch. Intelligent Agents.

Appendix A. UML Notation

In order to have a self-contained document, we present in this section a brief introduction to the Unified Modeling Language (UML) notation.

UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The Unified Modeling Language includes a set of graphic notation techniques to create visual models of object-oriented software-intensive systems.

In this small introduction of UML, we will just describe the basic notation, which is usually enough to model semantic systems.

The first key concept in a semantic model is the concept of Class.

Class: A class is a type that has objects as its instances OMG (2011). Classes have attributes (features) and operations (functionality) and participate in inheritance hierarchies. Figure A.9 shows the definition of two basic classes: *Share* and *Circle*.

In order to model inheritance (subclasses/especialized classes) in an UML-model, the generalization relation is used.

Generalization: A generalization is a taxonomic relationship between a more general classifier and a more specific classifier OMG (2011). A generalization is used to express inheritance and it is drawn from the specific classifier to a general classifier. Thus, the specific classifier indirectly has

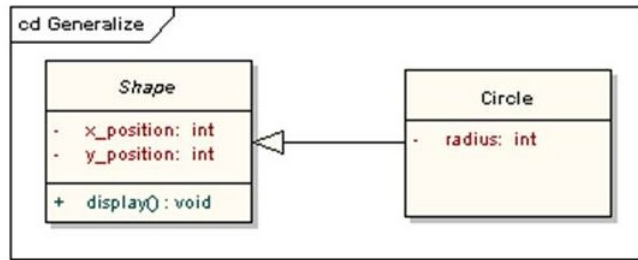


Figure A.9: Generalization between classes Sparx-Systems (2012).

features of the more general classifier. Figure A.9 shows a parent class *Shape* generalizing a child class *Circle* Sparx-Systems (2012).

A basic form for defining semantic relations is by using UML-associations.

Association: An association specifies a semantic relationship that can occur between typed instances OMG (2011), this implies that two model elements have a relationship. This connector may include named roles at each end, cardinality, direction and constraints. Association is the general relationship type between elements. In Figure A.10, a relation between the class *Team* and the class *Player* is depicted. We can see that in this association there exist cardinality constraints.

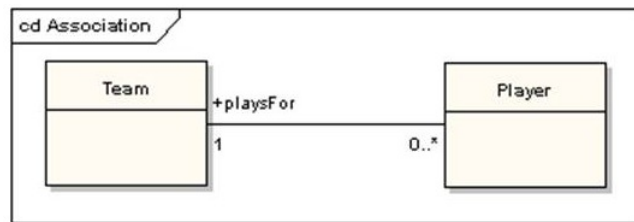


Figure A.10: Association between two classes Sparx-Systems (2012).

UML notation is also able to capture aggregations.

Aggregation: Aggregations are used to depict elements which are made up of smaller components. Aggregation relationships are shown by a white diamond-shaped arrowhead pointing towards the target or parent class Sparx-Systems (2012).

A stronger form of aggregation - a composite aggregation - is shown by a black diamond-shaped arrowhead and is used where components can be included in a maximum of one composition at a time. If the parent of a composite aggregation is deleted, usually all of its parts are deleted with

it OMG (2011). Nevertheless, a part can be individually removed from a composition without having to delete the entire composition. Compositions are transitive, asymmetric relationships and can be recursive Sparx-Systems (2012).

Figure A.11 illustrates the difference between weak and strong aggregations. An address book is made up of a multiplicity of contacts and contact groups. A contact group is a virtual grouping of contacts; a contact may be included in more than one contact group. If you delete an address book, all the contacts and contact groups will be deleted too; if you delete a contact group, no contacts will be deleted Sparx-Systems (2012).

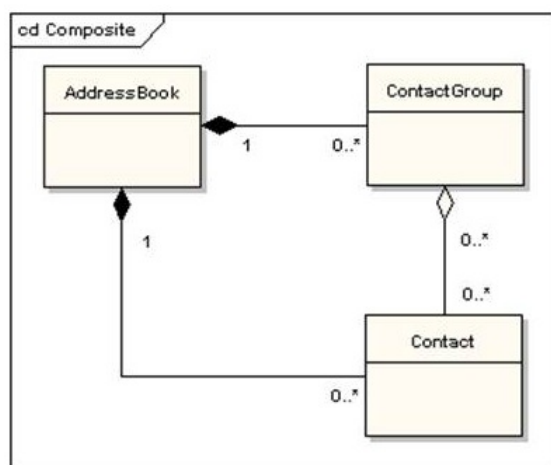


Figure A.11: Aggregation between classes Sparx-Systems (2012).