Journal of Ambient Intelligence and Smart Environments 12 (2020) 457–474
DOI 10.3233/AIS-200578
IOS Press

# Integrating IoT platforms using the INTER-IoT approach: A case study of the CasAware project

Gianfranco E. Modoni [a,*], Enrico G. Caldarola [a], Nicola Mincuzzi [a], Marco Sacco [a], Katarzyna Wasielewska [b], Paweł Szmeja [b], Maria Ganzha [b,c], Marcin Paprzycki [b,d] and Wiesław Pawłowski [e]

[a] *Institute of Intelligent Industrial Systems and Technologies for Advanced Manufacturing, National Research Council, via Lembo 38F, Bari, Italy*
*E-mails: Gianfranco.Modoni@stiima.cnr.it, enrico.caldarola@stiima.cnr.it, nicola.mincuzzi@stiima.cnr.it, marco.sacco@stiima.cnr.it*

[b] *Systems Research Institute, Polish Academy of Sciences, Newelska 6, 01-447, Warszawa, Poland*
*E-mails: Katarzyna.Wasielewska@ibspan.waw.pl, Pawel.Szmeja@ibspan.waw.pl*

[c] *Department of Mathematics and Information Sciences, Warsaw University of Technology, Koszykowa 75, 00-662, Warszawa, Poland*
*E-mail: M.Ganzha@mini.pw.edu.pl*

[d] *Department of Management and Technical Sciences, Warsaw Management University, Kawęczyńska 36, 03-772, Warszawa, Poland*
*E-mail: Marcin.Paprzycki@ibspan.waw.pl*

[e] *Faculty of Mathematics, Physics, and Informatics, University of Gdańsk, Wita Stwosza 57, 80-308, Gdańsk, Poland*
*E-mail: w.pawlowski@inf.ug.edu.pl*

**Abstract.** *CasAware* is an Ambient Assisted Living platform, developed within an Italian research project, with the aim to improve the level of comfort and well-being of inhabitants of a house, while optimizing the energy consumption. A key feature, for successful realization of such a platform, is its capability to interoperate with other IoT platforms, which can augment CasAware with additional services. Indeed, this capability facilitates smooth communication between CasAware devices and external devices connected to other IoT platforms, thus allowing efficient exchange of messages among them. However, such integration is hindered by the heterogeneity of data models used in different platforms, which is also related to lack of common standards. In order to realize integration needed for CasAware, this paper presents an approach which exploits results of the INTER-IoT project. Specifically, the INTER-IoT methodology and a set of software tools for achieving IoT interoperability are applied. In the presented study, it is shown how the INTER-IoT based approach can facilitate interoperability between CasAware and two other platforms, which use completely different data models.

Keywords: Internet-of-Things (IoT), CasAware, smart home, IoT platforms, INTER-IoT, semantic interoperability

## 1. Introduction

With the rapidly growing number of devices that produce and consume data, Internet of Things (IoT)

[*] Corresponding author. E-mail: Gianfranco.Modoni@stiima.cnr.it.

is not reducible to simply connecting these devices to the Internet. The major challenge that must be faced to exploit IoT in its full potential, consists of linking *things* (*IoT artifacts*) into synergistic ecosystems, where artifacts are endowed with virtual and physical components, which *work together* with a high degree of interoperability [2,9]. Under these conditions, success of any "generic artifact" depends on its capability to be interoperable with others. This capability can be exploited by the artifacts in various contexts, e.g. to pool their functionalities to deliver higher-level services [14]. However, such capability is hindered by the fact that artifacts found across IoT ecosystems come from various vendors, which usually adopt different (incompatible) communication interfaces, software stacks, operating systems, data models, and hardware [39]. This problem is worsened by a lack, in the IoT domain, of globally established standards that would be accepted by, at least, a majority of artifacts. In order to enable needed interoperability, it is crucial to identify valid methods of making things more cooperative and collaborative, providing them with capabilities to exchange information in a meaningful way (i.e., making it semantically understood by communicating parties) [40].

This research addresses current IoT needs by starting from, and extending, a preliminary paper of the same authors [38]. Specifically, it focuses on the integration of the CasAware platform [35] with two other platforms into an IoT ecosystem. CasAware represents the main outcome of an Italian research project, funded by the Lombardy region, and consists of an Ambient Assisted Living solution, which aims at optimizing energy consumption, within a household, exploiting cooperation among domestic IoT devices. The first platform to be integrated with CasAware is a meteorological observation platform, allowing to adjust the living environment parameters (e.g., indoor temperature, brightness, etc.) in response to the available meteorological data. The second integration involves a train information platform, which reports eventual delays to the train users. In addition, many other usage scenarios of the CasAware solution require interoperability with other platforms [35]. In one of these scenarios, CasAware allows to optimize the energy consumption and reduce the risk of unexpected power-cuts, also thanks to the CasAware's integration with the platforms of utility companies. In another scenario, CasAware enables automatic reordering of basic necessities when these are depleted, through the integration with an e-commerce platform. Moreover, more than one scenario requires to trace the indoor and outdoor movements of home inhabitants and for this reason it is essential to integrate the CasAware with specific Real Time Location System (RTLS) platforms.

Integration of CasAware with other IoT artifacts leverages, in this work, methodology, tools, and framework of the European research project INTER-IoT [20]. INTER-IoT aims at designing, implementing and testing a set of tools, along with a methodology that helps to achieve interoperability among IoT artifacts (platforms/systems/applications), on different layers of the technology stack (devices, network, middlewares, applications and services, data and semantics). Within INTER-IoT, various IoT platforms, from multiple application domains, originating from pilots and open call projects, have been integrated into the INTER-IoT-enabled ecosystems.

In what follows, it is discussed how the CasAware platform can utilize advantages brought by the INTER-IoT project, i.e.:

1. exchange of meaningful information between plugged artifacts in a seamless and smart way;
2. simplified communication between client and CasAware back-end services, by exploiting tested, robust communication mechanisms provided by the INTER-IoT implementation;
3. smooth integration of data between platforms at the semantic level, enabled by adopting a simple and efficient alignment format and translation mechanisms.

In summary, while our earlier work [38] introduced a preliminary design of the motivating scenario and an initial proof of concept, limited to a simple integration of small batch of information, this paper describes and implements the overall approach in order to allow exchange of a complete set of information among the different IoT platforms. The approach includes: (a) ontologies at the base of the three involved platforms and the Central Ontology which reunifies and integrates the previous ones; (b) a software Bridge component for each IoT platform, that enables application level integration of the platforms; (c) alignments between the three IoT platforms and the Central Ontology.

The remainder of the paper is structured as follows. Section 2 outlines state of the art of interoperability between IoT platforms. Next, in Section 3 an overview of the INTER-IoT approach followed in this work is provided, while Section 4 describes the CasAware system with its architectural levels, paired with a usage scenario that provides context to the application of the

INTER-IoT methodology. Section 5 describes the data models underpinning the semantic layer of considered scenario, while in Section 6 it is provided a description of the integration, within the usage scenario, of CasAware platform with other INTER-IoT-enabled components/products. Finally, Section 7 outlines conclusions and discusses future research directions.

## 2. Interoperability of IoT artifacts – state of the art

Ideas concerning possible solution to the problem of achieving interoperability of IoT artifacts evolved over time. According to [17], the first wave of IoT application emphasized connecting sensors, interfacing with physical-world using lightweight communication protocols such as CoAP (Constrained Application Protocol) [8], and XMPP (Extensible Messaging and Presence Protocol) [45], mainly within *smart city* domain. Moreover, the traditional Internet Representational State Transfer Protocol (REST) idea has been used for similar applications, where event-centric approaches had been implemented to reduce number of transmitted messages [18].

The second wave brought the idea of "smart objects", i.e., devices that incorporate certain degree of intelligence, making them able to "understand" the *environment* and react to external stimuli it provides. With growing real-world awareness, smart objects provide support for increasingly complex solutions [31]. In industrial applications, for example, existing physical objects, like containers and tools, as well as procedures, like, e.g., quality control, have been converted into smart objects, equipped with embedded sensors, wireless connectivity, and computational capabilities. Such smart objects are to be able to communicate, interpret sensor data, make decisions, and cooperate with each other. With the introduction of smart objects, the Internet of Things vision became reality [7,36].

Finally, the third wave involves use of *semantically enriched* data, acquired from *heterogeneous* sources in a, possibly, *multi-domain/cross-domain* environments. The need for semantics became obvious, once the IoT domain started getting congested with multiple platforms/applications using different (essentially incompatible) communication protocols and data models [5]. Due to proliferation of vendor-specific solutions, many organizations have been attempting to promote standardization, in order to enable/guarantee interoperability between applications. One of the first initiatives, in this direction was the EU funded OpenIoT project [48] which focused on developing open source middleware for IoT interoperability, using linked sensor data. At the heart of OpenIoT lies the old revision of W3C Semantic Sensor Networks ontology (SSNX) [52], which provides a common model for representing physical and virtual sensors. It also uses several well-known standard vocabularies and ontologies (e.g. PROVO provenance ontology [53], LinkedGeoData [54] and Basic Geo Vocabulary [55], LSM live sensor data management, etc.), as well as custom pilot-specific ontologies to model the necessary concepts [50]. FIWARE [44] is another, EU sponsored, platform for IoT, enabling a market-ready open source solution, which combines components that enable the connection of IoT with Context Information Management and Big Data services in the Cloud. On the other hand, IoT solutions originating from the private sector include those developed by the AllSeen Alliance, in conjunction with the Open Connectivity Foundation (OCF). AllJoyn [3] is an open source software framework that enables devices and applications to discover and communicate with each other, at the same time freeing developers from the details of the transport layer, the manufacturer-specific differences, and so forth, when they develop IoT applications. Numerous approaches have also been used to align and integrate different communication protocols or data models, instead of focusing on creating a *single* standardized one [11,33].

By leveraging theoretical and technological approaches, developed within the *Semantic Web*, data produced and processed by IoT artifacts progressively becomes semantically-enriched or, even, semantics-based. In the case of raw sensor data, for instance, the enrichment usually refers to contextual information, compliant with corresponding data models. For instance, for sensor data it can be mentioned: i) OGC's Sensor Web Enablement (SWE) [41], established by the Open Geospatial Consortium, which includes the following important specifications: Observation & Measurement (O&M), Sensor Model Language (SensorML) and Sensor Observation Service (SOS); ii) Semantic Sensor Network (SOSA/SSN) ontology [52], developed by W3C that provides standard for modeling sensor devices, sensor platforms, knowledge of the environment and observations; iii) Semantic Sensor Observation Service (SemSOS) [28], and iv) Smart Appliances REFerence ontology (SAREF) [19]. To facilitate "semantic-enrichment" a number of methods/approaches for different data formats and IoT-related *ontologies* emerged [21]. Specifically, these methods focus on converting implicit semantics that are contained in other schemas (e.g. XSD, JSON schemas, relational database schemas) into explicit semantics i.e. ontologies and vocabularies.

Although the aforementioned semantic methods contribute to enhance integration of IoT artifacts (at semantic level), the full IoT interoperability challenge is still far from being solved. Indeed, it cannot be expected that the all the IoT data producers and consumers will ever agree on using the same shared semantic model and, on the other hands, there will ever a leading semantic data representation that will dominate the others [22,51]. The result is an increasing data fragmentation among vertically-oriented artifacts that needs to be overcome. With the growth of the number of embedded ambient devices connected in the smart living environments, this need is becoming more and more prominent [42]. For this reason, it is essential to address the enhancement of semantic interoperability among the available IoT artifacts, thus allowing to facilitate the understanding of messages exchanged among them [17,23]. In order to ease this interoperability, the differences among the ontologies on which the different IoT artifacts are founded have to be mediated and reconciled. To realize this reconciliation, this study adopts, within a real use case of the CasAware project, the solution proposed by the INTER-IoT consortium [20], which leverages the alignment pattern to enable the semantic translation among heterogeneous ontologies. Specifically, the overall aim of this study within CasAware is allowing the exchange of a complete set of information among the different IoT platforms. Under these conditions, for the CasAware platform, it is also crucial that interoperability is considered at different levels of technology stack e.g. middleware and semantics, as provided by INTER-IoT approach. Another important aspect which influenced the adoption in CasAware of INTER-IoT approach is the latter's capability to not to deploy another IoT platform for enhancing interoperability, but rather add components that enable interoperability without changing the internal processes in a given IoT artifact. In addition, the INTER-IoT approach promotes in CasAware the reuse of the information scattered across the different IoT platforms (e.g., reuse of overlapping data models, reuse of the alignments if different platforms send messages with similar semantics, etc.). On the other hands, the CasAware project provides a valid context for the evaluation of the INTER-IoT solution, by allowing to focus in particular on the problem of identifying and representing alignments.

It should be noted that, in recent years, several projects addressing different aspects and levels of interoperability have been analyzed and realized in fields such as smart city and smart environments. The uptake of initiatives in this field proves that work done in CasAware is in line with current trends. In this regard, one of the projects is SmartSantander [46] utilizing results of SENSEI and WISEBED EU projects. The Santander city was equipped with more than 12 000 sensors that collect data that are later accessible for researchers and applications creators. The project's objective was to provide experimental facility that enabled to test cutting edge technologies. SmartSantander itself acted as en enabler by deploying the necessary and costly infrastructure. FIESTA-IoT project [1,13,27] aimed at federating a large number of testbeds, in order to offer experimenters the unique experience of dealing with a large number of semantically interoperable data sources. The idea to provide semantic interoperability was based on reusing generic ontologies and extending them wherever required to meet the requirements identified for the federation of IoT testbeds. Additionally, initiatives such as IoT-A [6] aimed at unifying different ontologies in the IoT landscape. In contrast, INTER-IoT project provided ontology-agnostic mechanisms to implement semantic interoperability by semantically translating data streams. In INTER-IoT the modular and extensible GOIoTP ontology for IoT domain was proposed but is not required for the proposed semantic translation solution to work.

Another example is the OrganiCity project [4,58] which delivered an implementation of an Experimentation as a Service framework, including a toolset that allowed developing, deploying and evaluating smart city solutions. The proposed services, among others, addressed the issue of access to open data. The focus of the project was put on providing support mechanisms enabling urban data and IoT experimentation, whereas the CasAware project with INTER-IoT components is focused on deploying a system interoperating with other IoT platforms.

## 3. INTER-IoT project overview

Recently completed EU H2020 INTER-IoT project [20] aimed at the design and implementation of a set of tools and a methodology, dedicated to achieving interoperability between heterogeneous IoT artifacts. Interoperability mechanisms work at the various layers of

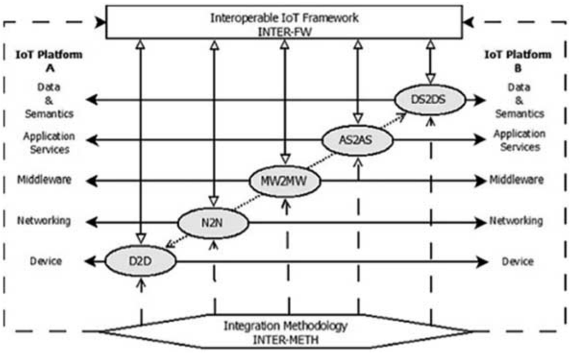

Fig. 1. INTER-IoT conceptual framework.

the hardware/software stack, i.e. devices (D2D), network (N2N), middlewares (MW2MW), applications and services (AS2AS), data and semantics (DS2DS); see Fig. 1.

The INTER-IoT solution enables creation of IoT ecosystems, in which uni- or bi-directional communication can be established at *any* layer of the hardware-software stack. Additionally, the solution offers unified INTER-API that allows to configure and interact with each layer's components. In the context of CasAware, (syntactic and semantic) integration is being performed at MW2MW (Middleware to Middleware) layer, where INTER-IoT component INTER-MW enables the data exchange among platforms e.g. CasAware and the other two platforms considered in this contribution: *Viaggiatreno* and *OpenWeather*. Within INTER-MW, the semantic interoperability is achieved by utilizing the IPSM (Inter Platform Semantic Mediator) that applies an alignment-based translation of semantically-annotated messages [22]. Here, it should be noted that, while in the CasAware integration IPSM is used internally by the INTER-MW middleware, IPSM can also be used as a standalone software component, which makes it particularly useful in building ecosystems consisting of multiple IoT artifacts.

It should be stressed that the INTER-IoT interoperability approach *does not* require changes to the connected platforms/artifacts (e.g. the CasAware platform, and the remaining two that are to be joined with it). To achieve interoperability, an INTER-MW connector called *Bridge* needs to be implemented, from a provided template. Part of the Bridge functionality is to perform syntactic translation (i.e. converting messages to the JSON-LD format, which is used by INTER-IoT internally and annotating them with an ontology – platform-specific or standard). Depending on the needs of the integration process, there can be a one-way or a two-way syntactic translation of data formats. Specifically, if IoT artifact only publishes messages, a one-way translation may be required, otherwise translation in both directions should be implemented (as two separate processes). Let us underline that the syntactic translation does not dictate use of any specific ontology.

Every INTER-IoT JSON-LD message consists of two RDF graphs *metadata* and *payload*, where the payload represents the "core content" of the message, and metadata is used to describe the message itself, e.g. who sent it, and when. Therefore, for instance, if a source platform uses XML based on XSD, then the Bridge should translate the message to JSON-LD, where the *payload* has the same information content annotated with the source platform's ontology/vocabulary [24]. Before integration, XSD for IoT artifact should be mapped onto IoT artifact dedicated RDF. If the platform natively supports RDF, then INTER-IoT messaging libraries may be used directly to use platform RDF in message payloads. Otherwise, data schema used by the platform should be "lifted" to ontology, that can be later used to annotate messages leaving the Bridge.

Internally, within INTER-MW, the semantic translation to/from a *Central Ontology* (CO) (INTER-IoT model, extendable for specific deployments) and source/target artifact's ontology is performed by the IPSM component. The translation is configured with *alignment* files, written in the IPSM-AF format, i.e. mappings between graph structures in source and target ontologies [50]. INTER-IoT communication is based on a publish-subscribe paradigm, consequently there can be many publishers and subscribers for any semantic translation channel. Alignments defining rules for translation between source/target semantics and CO can be reused for entities using the same semantics.

The CO serves as an intermediate data model (according to the translation architecture proposed by IPSM), and is designed to have modular structure. The "core", in most INTER-IoT deployments, is the Generic Ontology for IoT Platforms (GOIoTP) [50]. However, since the IPSM is agnostic to the ontology that is being used as the central one, any other ontology can be used here. Additionally, it can be extended with deployment specific modules, e.g. medical, logistics, meteorological, etc. In this way, the IPSM can handle translations of messages in any domain-specific context. Alignments are defined between pertinent artifacts ontologies (their parts that are actually used in considered messages) and specific modules of the central ontology. Here, let us note that the proposed approach does not require full ontology-to-ontology translation. Alignments involve only those ontology fragments that are needed for a given message exchange to be understandable to participating artifacts. In this way, while being ontology based, the proposed approach is more flexible and efficient, and easier to get started with [25].

## 4. CasAware project overview

The CasAware project belongs to the area of, broadly understood, ambient assisted living. It aims at improving the level of comfort and well-being of inhabitants, while optimizing energy consumption and improving security. Specifically, it supports developing a context-aware system, which monitors the dwellers' behaviour, in order to provide customized services related to their safety in the house, appliance management, and energy consumption management [35]. Customized services are also to be provided by integration of CasAware with other, external, IoT platforms, which provide further information to the CasAware ecosystem.

### 4.1. *CasAware architecture*

The proposed system leverages combined exploitation of various technologies, ranging from IoT to Big Data and the Semantic Web. The system is currently being developed at the STIIMA CNR's IoT Living Lab in Lecco [49], and is to be tested with real inhabitants (both able-bodied and impaired). Specifically, the CasAware system is structured into five layers, as depicted in Fig. 2 [35].

The first layer is a physical network of interconnected and interacting devices. Connection between these devices and people can be realized leveraging active transponders that are suitably equipped with sensors connected to various receivers, to transmit acquired home telemetry, which traces in near real time the position of inhabitants of the house.

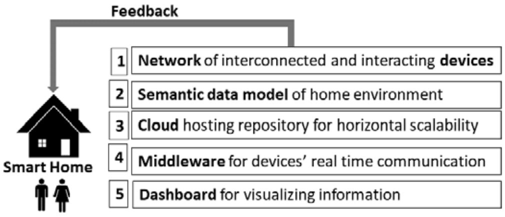
Feedback
Smart Home
1 Network of interconnected and interacting devices
2 Semantic data model of home environment
3 Cloud hosting repository for horizontal scalability
4 Middleware for devices' real time communication
5 Dashboard for visualizing information

Fig. 2. A layered-based architecture for CasAware project.

The second layer is a semantic data model that formally represents knowledge of home environment and behaviors of its inhabitants. Furthermore, the semantic-based approach enables use of reasoning to infer new insights from already known facts. The semantic model is described, in detail, in Section 5.1, focusing on those models that are important for application of INTER-IoT approach.

The third layer is a cloud platform that guarantees horizontal scalability of the whole infrastructure. The cloud hosts the semantic database (Stardog[1]), where the semantic model is persisted (both instances and corresponding meta-model), exploiting the exposed SPARQL engine [39]. The third layer leverages also the reasoner included in Stardog and a set of inference rules that, together, allow to entail and extract new knowledge.

The fourth layer is a publish/subscribe semantic middleware (SM) that leverages the semantic model (from the second layer) to provide services for events notification, among interested devices [37,40].

Finally, the fifth layer is represented by the graphical user interface (Dashboard) for visualizing pertinent information.

### 4.2. *CasAware integration: Motivating scenario*

In [35], authors described typical scenarios involving CasAware IoT platform. The first one highlights well the benefits of connecting the CasAware platform to other IoT solutions. It can be summarized as follows.

Let us consider Francesco, who is 42 years old and suffers from hypotension. CasAware system is aware that Francesco gets up every morning at 6:50 and leaves home to work at 7:35. It also knows that in regular conditions, it takes Francesco about 45 minutes to reach work, by walking 850 meters to the nearest train station, and then taking a train that stops at a station that is right next to the building where he works.

The idea behind this work is that the implemented CasAware platform could better support Francesco's daily acts exploiting the services offered by two other IoT platforms. Indeed, the latter would allow CasAware to receive real-time information regarding train and weather conditions (*Viaggiatreno*[2] for train infor-

[1] https://www.stardog.com/
[2] http://www.viaggiatreno.it

mation, and *OpenWeather*[3] for weather conditions). Thus, CasAware can estimate potential slowdowns on Francesco's usual path to work (e.g. due to heavy rain, delay of the train, etc.), which in turn can cause a delay of $N$ minutes in his arrival at work. In this case, according to acquired information, CasAware will send information to the alarm clock and will wake Francesco $M$ minutes earlier than usual (to catch train that leaves $M$ minutes earlier). Similarly, the coffee machine will start $M$ minutes earlier and the coffee will be ready as soon as Francesco enters the kitchen. In addition, earlier ignition of the heating system will be needed, to ensure a warm environment when Francesco rises. The system will then be switched off, as soon as Francesco will leave the house, thus optimizing gas consumption. When Francesco will try to open the door to leave the house, the door is not going open if he has not taken the medicine for the hypotension. Finally, in case of rain or snow, the umbrella standing near the door will blink (thanks to the integration with *OpenWeather*), reminding Francesco to take it with him.

The above scenario advocates the need to enhance the interoperability of CasAware with the other two platforms. Indeed, compared to CasAware, Viaggiatreno and OpenWeather use completely different data model representation and for this reason a translation mechanism is needed. While it could be possible to develop one-to-one translators for each platform that the CasAware is to communicate with, such approach would not scale with a growing ecosystem and, potential, more complex interactions. Here, it should be recalled that in Section 1 it was stipulated that the CasAware platform should, in the future, be integrated with a platform of an utility company, an ecommerce platform, and an RTLS platform. This would introduce three additional one-to-one translations. To avoid such rapid growth of number of translators (with the problem being not only their number, but also their long-term maintenance), mechanism based on the INTER-IoT IPSM has been selected to facilitate translation between data models. With IPSM, messages are translated to Central Ontology which later enables multiple subscribers to consume messages from one downstream channel. On the other hand, if multiple publishers send messages with similar semantics, one upstream alignment can be prepared and reused to consume messages originating at different artifacts.

In the next sections, an overview of data models used for the three platforms is provided, starting from the CasAware ontology. Moreover, the alignments needed to realize the proposed scenario will be elaborated, including their application within the IPSM.

## 5. Ontologies modelling the motivating scenario

The semantic models used in the CasAware platform are expressed as a set of ontologies, which are "formal, explicit specifications of a shared conceptualization" [26]. For the formalization of the ontologies, the standard languages of Resource Description Framework (RDF) [56], the Web Ontology Language (OWL) [57] and the Semantic Web Rule Language (SWRL) [29], endorsed by the W3C, are used. Each ontology is represented as a list of triples, expressed under the form of subject, predicate, and object.

Specifically, the motivating scenario above described is represented by four ontologies: one ontology for each platform, and one used as CO. The scope of the latter one is allowing the semantic translation between platforms. The design of all ontologies has been carried out with respect to the following guidelines:

- Identify the content of each platform ontology regarding the domain concepts and logical links to be added in the ontological model and group them in macro-categories or modules;
- Evaluate possible reuse of third-party ontology modules – prioritizing those already included in the GOIoTP ontology – by searching for suitable existing open modules in publicly available repository such as Linked Open Vocabulary et similia [34];
- Identify the *conversations* that will take place within the demonstration scenario, in terms of communication acts and of information shared among the platforms involved in the scenario.

The subsequent sections describe in detail how these steps have been applied to obtain each ontology underpinning the platforms used in the demonstration scenario.

### 5.1. CasAware Ontology

Figure 3 shows modules belonging to the CasAware Ontology (represented as hexagons while dotted lines links modules with their significant classes). Here, the

[3] http://openweather.com

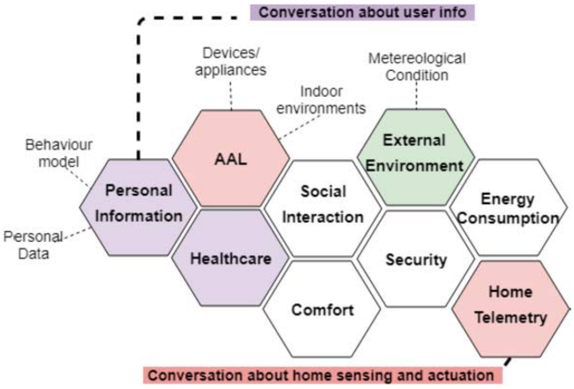

Fig. 3. CasAware Ontology modules.

key ones are: (1) *Personal Information*, which contains classes and relations linked to the personal data of inhabitants, or their behaviors (activities, habits, etc.); (2) *Ambient Assisted Living* (AAL) module, which contains classes and relations about devices/appliances, or similar objects, deployed throughout the house, (3) *External Environment* module, which contains classes and relations related to the meteorological conditions; (4) *Home Telemetry*, which encapsulates concepts and relations related to the characteristics of data streams originating from ubiquitous sensors, e.g., data format, frequency, volume, etc.. CasAware Ontology contains also additional modules such as: *Health-Care*, *Social Interactions*, *Security*, *Comfort* and *Energy Consumption*.

Recognizing that reuse is a good practice in ontology design [12], CasAware Ontology has been designed by incorporating existing data models and/or ontologies, some of which have already been included in the GOIoTP. Specifically, CasAware uses: (1) the Semantic Sensor Network (SSN) ontology, which describes sensors and their observations, the involved procedure and features of interest, and, in turn, uses SOSA (Sensor, Observation, Sample, and Actuator) ontology for its elementary classes and properties [15]; (2) the Dublin Core ontology, a set of metadata elements for cataloging generic items or resources [32]; (3) the FoaF (Friend of a Friend) ontology, which collects a variety of terms describing people, groups, documents, etc. [10]. The latter modules are inherited from the GOIoTP ontology, while, those specifically related to CasAware are: the Virtual Individual Model (VIM) [47] for *Personal Information*; the Ontology Modeling for Intelligent Domotic Environments (Dogont[4]), which supports device/network independent description of houses, including both controllable and architectural elements; and, finally, the Smart Appliances REFerence (SAREF) ontology [16], which is a shared model of consensus that facilitates the matching of existing assets (standards, protocols, data models, etc.) in the smart appliances domain.

### 5.2. *Viaggiatreno Ontology*

Viaggiatreno platform is an IoT platform gathering and publishing public/private transportation data. In particular, it is focused on providing real-time data about train paths, relative timetable and eventual delays. In addition, it exposes a REST API that provides structured information.

In Fig. 4 an excerpt of the Viaggiatreno Ontology modules, represented as hexagons, is shown. The modules contain in turn classes (in figure the significant one are linked to the modules through dotted lines) and relations concerning means of public transportation, transportation services, time tables, etc.. Since the Viaggiatreno Ontology will be used in the Section 6.2 to provide an example of alignment, its excerpt is reported in Fig. 5. This ontology is built upon the *TrasportationService* module, which represents the core of the platform, and allows to interconnect different modules of ontology, in order to model every type of public/private transportation service. A direct subclass of *TrasportationService* is *TrainService*, which is a basic class encapsulating all information about trains on the Italian national territory. This class is tied to additional modules and classes, which enrich its definition. These are: (1) *Route* class, which represents the train path inher-

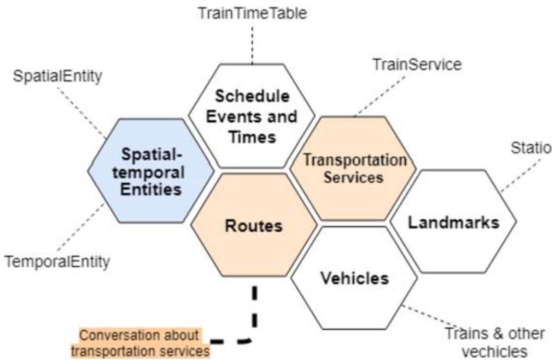

Fig. 4. Viaggiatreno Ontology modules.

[4] http://www.cad.polito.it/pap/exact/iswc08.html

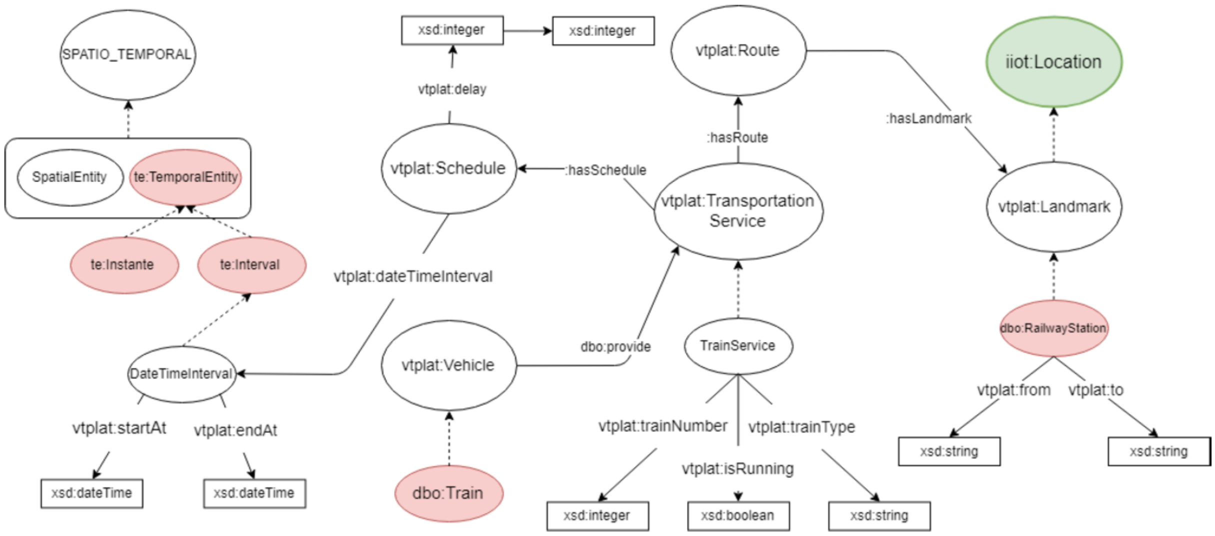

Fig. 5. Viaggiatreno Ontology excerpt.

ent to the user trip, where the departure and arrival railway stations are provided; (2) *Landmarks*, which defines and describes interest points along the train route by suitable classes/properties; (3) *Schedule* module, which represents the timetable relative to each train; finally, (4) *SpatioTemporal* module that describes each event relative to a train service in terms of space and time. In particular, the *SpatialEntity* and *TemporalEntity* classes have been designed, by including existing data models such as `Geo`[5] and `Time`[6] vocabularies. In addition, the ViaggiaTreno Ontology establishes the reuse of `DBpedia`[7] which allows to enrich the *Landmarks*, *TrasportationService* and *Vehicle* classes by reuse of entity such as `dbo:Station`, `dbo:RailwayStation` or `dbo:Train`.

### 5.3. OpenWeather Ontology

OpenWeather platform is an IoT platform for weather monitoring, which provides weather information, such as current weather conditions, forecasts and historical data, by exploiting REST Web services. Particularly, it encapsulates information about meteorological phenomena such as precipitation, fog, thunderstorm, etc., and the related physical variables that can be measured, e.g., temperature, atmospheric pressure, humidity and so forth.

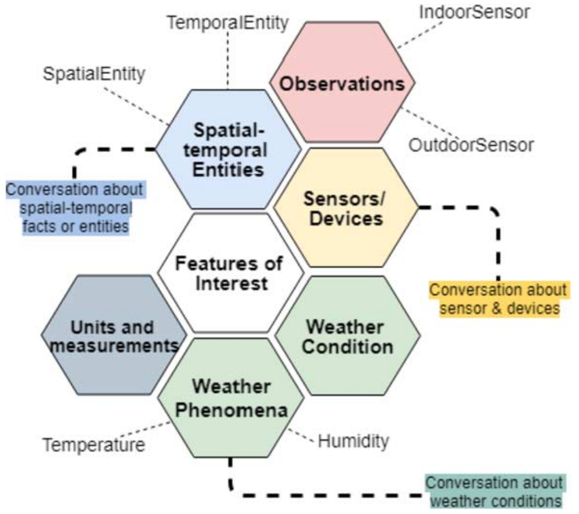

Fig. 6. OpenWeather Ontology modules.

In Fig. 6 different modules, included in OpenWeather, are shown represented as hexagons. One of these modules is the *Observation* module, which is the central hub able to provide the current weather in a specific city and in a given time instant. Another module is the *Sensor* module, which is based on the reuse of the Semantic Sensor Network (SSN) ontology and describes devices, agents (including humans), or software (simulation), involved in monitoring of specific weather conditions in a given geographic area. Each event is tied to different weather features, which are described in *FeatureOfInterests* module. The latter split into two subclasses: *Weath-*

[5] http://www.w3.org/2003/01/geo
[6] http://www.w3.org/2006/time
[7] https://wiki.dbpedia.org/

*erCondition* and *WeatherPhenomena*. The *WeatherCondition* class comprises the state of the atmosphere in terms of temperature, wind, clouds, and precipitation. The *WeatherPhenomena* class, instead, is defined by natural events that occur as a result of one or a combination of the water cycle, pressure systems and the Coriolis effect. Finally, each event is tied to the *SpatialEntity* and the *TemporalEntity* classes that have been designed, as for ViaggiaTreno Ontology, by including existing ontologies and allow describing weather conditions in terms of space and time.

### 5.4. *The Central Ontology*

The role of the CO is crucial in integration of the IoT platforms within the motivating scenario. In fact, it represents a "*pass-through*" ontology the scope of which is: (1) facilitating translation of semantic triples from one platform ontology to the others, by driving the creation of a partial alignment mapping for selected fragments of pertinent ontologies; (2) inheriting and (optionally) extending as many elements as possible from the GOIoTP ontology, thus subsuming it and leveraging Application Program Interfaces (APIs) provided by the INTER-IoT framework; (3) informing the model view of the client application used in the demonstration scenario (hereafter referred to as the *CasAware Dashboard*).

In order to achieve objectives of the project, the following rules have been stressed during the design phase of the ontology: (1) the CO must include concepts and semantics coming from all the platforms that must be integrated; (2) the CO must provide *handy* data to be easily used by the client application, e.g., it must simplify the semantics of the platforms ontologies, for the sake of clearness and usefulness of integrated data; (3) it must inherit as many "stub" concepts as possible from the GoIoTP ontology, which will subsume specialized classes from platforms ontologies. This way, the CO acts as an extended ontological module of the GoIoTP ontology and can leverage the INTER-IoT framework APIs. In Fig. 10 the role of the Central Ontology, as conceived here, is depicted.

Many of the concepts included in the CO describe features coming from the client application and elicited in the demonstration scenario, and concepts inherent to the GOIoTP Ontology. Indeed, the latter represents a starting point for the hub ontology design. It offers modular structure for describing entities commonly appearing in IoT contexts, mostly concerning the interoperability of various IoT artifacts (platforms, devices, services, etc). GOIoTP Ontology also includes a set of extension modules, dubbed the Generic Ontology for IoT Platforms Extended (GOIoTPex) scope of which is "filling" stub classes from GOIoTP with more specific classes, properties and individuals, required in the concrete instance and implementation in the INTER-IoT project.

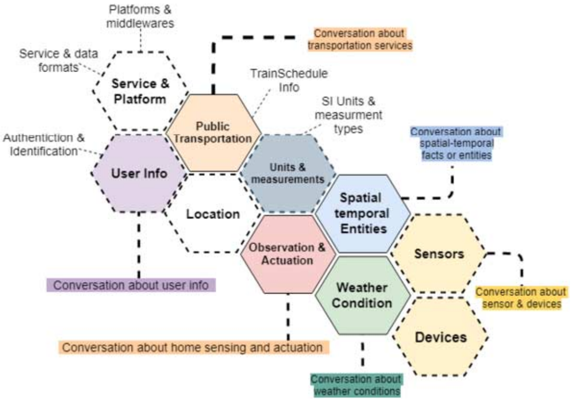
Platforms & middlewares
Service & data formats
Service & Platform
Conversation about transportation services
TrainSchedule Info
SI Units & measurment types
Public Transportation
Authentiction & Identification
User Info
Units & measurements
Conversation about spatial-temporal facts or entities
Spatial temporal Entities
Location
Observation & Actuation
Sensors
Conversation about user info
Weather Condition
Conversation about sensor & devices
Devices
Conversation about home sensing and actuation
Conversation about weather conditions

Fig. 7. Central Ontology modules.

The main modules included in the CO are the following (in Fig. 7 represented as hexagons): (1) *Device*, (optionally virtualized) hardware connected to IoT; (2) *Platform*, software platforms that host IoT devices; (3) *Observation (and actuation)*, information gathered by IoT entities from the world, or the intention to act upon it; (4) *User, human or software* that uses, or is a client of IoT entities; (5) *Location*; (6) *Service*.

In Fig. 8, an excerpt of the CO is provided with the upper level concepts and relations coming from the GOIoTP ontology. In this figure, the dashed ovals refer to the central ontology specific concepts while the green ovals to concepts inherited from GOIoTP (or GoIoTPex). Finally, the red ovals refer to classes imported by GOIoTP from third-party modules like sosa:Sensor or sosa:Platform.

The main concept within the CO is that of *Info Provider*. Different types of information providers are present in CO the most important ones being:

- `cwdash:AppliancesInfoProvider`;
- `cwdash:ThermoHygrometric ConditionsProvider`;
- `cwdash:NextTrainAvailable InfoProvider`;
- `cwdash:WeatherConditionProvider`;
- `cwdash:IndoorPositionProvider`.

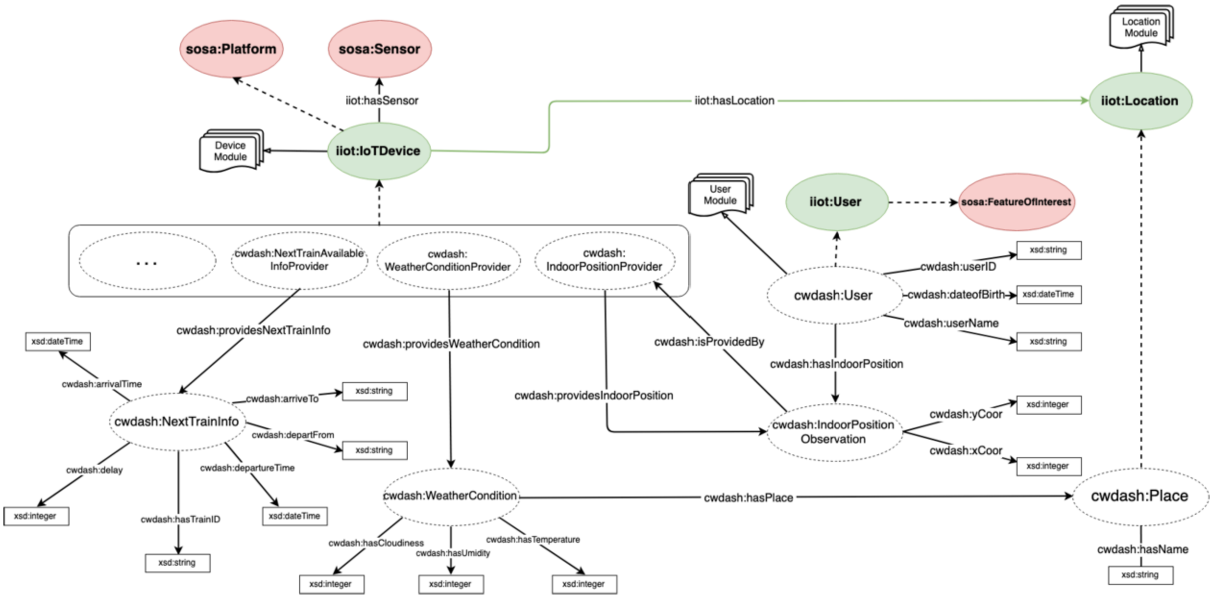

Fig. 8. Central Ontology excerpt.

All providers can be considered as abstract devices able to collect data from smart objects within the domestic environment, or from external web services, and provide them to the client application. They are defined as subclass of `iiot:IoTDevice` class, and inherit datatype and object properties from that class, e.g., `iiot:hasLocation`, which links `IoTDevice` and `iiot:Location`. Other concepts, specifically concerning the central ontology, are:

- `cwdash:IndoorThermometer`,
- `cwdash:IndoorHygrometer`,
- `cwdash:LockerSensor`,
- `cwdash:eBeacon`,

which inherit from `sosa:Sensor` class. Additionally, `cwdash:IndoorPosition` and `cwdash:User` are used to model real-time location of inhabitants in the domestic environment, while, `cwdash:NextTrainInfo`, `cwdash:WeatherCondition`, `cwdash:Place` represent bridge concepts between the client application and the external platforms (ViaggiaTreno and OpenWeather).

### 5.5. *Conversations within the demonstration scenario*

In Fig. 3, 4, 6, and 7, the bold dashed lines linked to the ontological modules represent the conversation channels established between the platforms and the CO. On meta-level, such channels represent content shared between the platforms and the CasAware Dashboard, at a specific domain. For example, all information about Transportation services and Routes in Viaggiatreno platform is mapped to the concepts and logical links just contained in the *Public Transportation* module of the central ontology. While, here, conversation channels are identified at an high-level, in Section 6.2 they are detailed at concepts and logical links level by highlighting concrete alignments.

## 6. Implementing the motivating scenario leveraging the INTER-IoT approach

This section describes activities carried out to implement the motivating scenario and in particular to integrate different IoT platforms. This integration process takes place both at an application level, by leveraging the INTER-IoT APIs, and at a semantic level, by exploiting the features of already mentioned IPSM component of INTER-IoT.

### 6.1. *Integration at the application level: Use of INTER-IoT APIs*

As reported in Section 3, the main idea of the INTER-IoT approach behind the integration at application level consists of creating a *Bridge* component for each platform, and integrating it into the INTER-IoT middleware. The implementation details about the

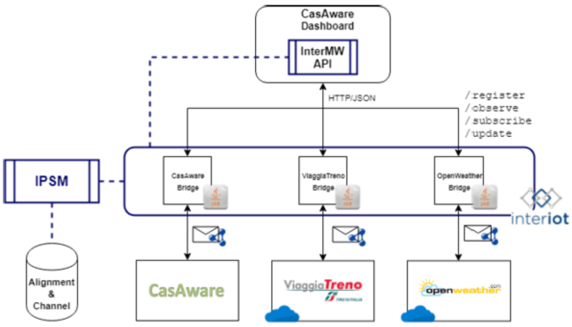

Fig. 9. The integration architecture of CasAware within Inter-IoT.

creation of the Bridge are out of scope of this work, but further explanations can be found in [30].

The Bridge acts as a *syntactic translator* transforming data formats from the ones used by IoT artifacts and the representation used within the INTER-MW middleware. Its implementation is based on a generic interface, which provides a structured template enabling easy development of new, platform-specific, instances. Once the *Bridge* is implemented for a specific platform (under the form of jar module), it enables the platform to perform basic operations by invoking RESTful methods. In particular, as shown in Fig. 9, every connected platform is able to invoke basic methods to establishing conversation between itself and the client application, e.g.: (1) *register*, in order to inform the INTER-IoT middleware about the presence of a new platform registered within the ecosystem; (2) *registerDevice* to register a specific device, with a unique ID, that will act as a data producer; (3) *observe*, the basic method the client application uses in order to get data from a device, (4) *actuate* to perform an actuation to a specific actuator device, and so forth.

The methods, implemented within the Bridge, include logic that is platform-specific and requires uni-/bi-directional communication with INTER-IoT middleware components (uni-directional communication takes place if only communication from the device, e.g. a sensor, to another platform is needed). In this way, on the one hand, a client application or device, deployed on some platform, can request or update information originating in CasAware via the REST services provided by the INTER-MW API. On the other hand, any device connected to CasAware can request information updates coming from any other, INTER-MW connected, IoT platform/artifacts. This last feature is particularly useful for the herein presented work. In order to exploit it, the CasAware Dashboard is in charge of registering itself and all the involved IoT platforms to the INTER-IoT middleware and instantiating devices, from which data will be observed, by exploiting the interfaces exposed by the platform's Bridge. Thus, the CasAware Dashboard is able to smoothly collect and integrate data coming from the various platforms, at the application level, disregarding different levels of heterogeneity characterizing them. The code of the three implemented Bridges is available within the *INTER-IoT github repository*.[8]

### 6.2. *Integration at the semantic level: Use of the IPSM*

As stated in Section 3, INTER-MW uses the IPSM to perform (streaming) semantic translation. While the syntactic translation is the responsibility of the Bridge, which implements a two-way change of data formats, the IPSM allows establishing interoperability at a semantic layer for all IoT artifacts connected to INTER-IoT framework. Figure 10 depicts the schema of alignments used in the motivating scenario. Here, triples coming from one platform ontology are aligned by IPSM with triples of each other ontology by exploiting the intermediate triples definitions contained in the CO.

As shown in Fig. 10, IPSM uses channels to perform translation, according to a common approach existing in the literature for *data integration* [43]. In such approach, the CO is considered a sort of "hub" or "pass-through" ontology, which allows converting triples from one source ontology to other target on-

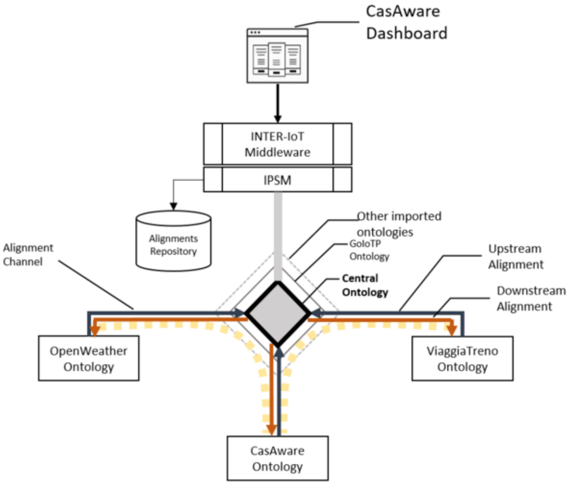

Fig. 10. CasAware Project alignments schema.

[8] https://github.com/INTER-IoT

tology without the need of creating a dedicated alignment for each pair of ontologies that are used in communication. For each platform ontology, two alignments are provided: the first one used to convert triples from that ontology to the hub (the so-called *upstream alignment*), and the second one to convert triples from the hub ontology to the platform ontology (the *downstream alignment*). It should be noted that if one of the connected artifacts uses semantic representation that is equivalent to the CO, semantic translation is not needed. Once a pair of alignments (upstream and downstream) is created for any platforms that want to talk to each other, translation channels are instantiated allowing to create a semantic conversion path between one platform to the other. For example, by using the upstream alignment from the OpenWeather to the Central Ontology, and the downstream alignment from the CO to the CasAware Ontology, a logical conversion link emerges from the OpenWeather to the CasAware Ontology.

The use of such an approach to ontology alignment, which involves the CO, presents several advantages: (1) decoupling the high-level domain-specific classes or axioms of each platform ontology from the application-specific axioms of the CO; (2) each domain ontology can remain self-consistent, and autonomous, avoiding incorporation of other axioms, not specifically related to the given platform itself. In this way, platform ontologies may incorporate axioms originating from well-know ontologies, existing in the literature, or may import them as a whole, by leveraging the reuse of data models; (3) a better modularization of the ontology design, which allows to minimize possible inconsistencies, duplication or redundancy, and improving readability and clearness of the ontology models.

As described above, the IPSM component can use upstream and/or downstream alignments for each platform-to-hub path and vice versa. The translation can be uni-directional or bi-directional depending on the specific context/application. In fact, it may be possible to have only one-way translation, with no need to send messages back. In the herein presented motivating scenario, the translation is uni-directional (downstream) as the data flows from the IoT platforms towards the CasAware Dashboard (Fig. 10).

The process of creation of alignments is, mostly, manual and involves the following steps: (1) identify messages exchanged between the platforms and the CasAware Dashboard by checking if some overlaps emerges between concepts and properties of the platform ontologies and the CO (this stage has already been accomplished at design time, when the platform ontologies have been constructed, see Section 5); (2) define the effective mappings, which involves in turn: (a) the identification of platform ontology concepts that directly map to concepts of the CO; (b) typify concepts according to the GOIoTP ontology stub classes (this in order to use INTER-MW Messaging API, e.g., typify a device as rdf:type IoTDevice); (3) adapt and simplify data properties from platform ontologies to application-oriented properties, in order to be used easily at the application level. Note that while the procedure definitely requires human in the loop, a tool that will assist ontology engineers in creating and verifying correctness of alignments is currently under development.

The above procedure has been applied to ontologies of all involved platforms (CasAWare, ViaggiaTreno and OpenWeather) vis-a-vis the CO. For the sake of brevity, the remainder of this Section will focus on the alignment between the Viaggiatreno and the CO ontological entities (classes and object relations). As an example, the XML fragment in Listing 1 shows an excerpt of alignments between the two ontologies, serialized in the IPSM-AF format. Since the alignment application acts at a sub-graph level, i.e., at a triples level, the single mapping in the listing is referred to a *cell*. This latter has a source and target subgraph pattern as specified in *entity1* and *entity2* elements. The alignment in the example includes three mapping steps, each of which is responsible for a specific part of the process, and will be executed sequentially.

In Fig. 11 the mapping specific to the first cell (lines from 9 to 30 in the xml listing) is graphically depicted. As shown in the figure, different entities from the Viaggiatreno ontology are mapped to a single entity in the CO. Specifically, `dbo:Train`, `vtplat:TrainService` and `vtplat:Schedule` from the source ontology contribute to feed two datatype properties at the target ontology (`cwdash: numTrain` and `cwdash:delay` respectively, which are linked to `cwdash:NextTrainAvailableInfo`).

The second step of the mapping defines the correspondence between `dbo:railwayStation` entity from the source ontology and the `cwdash:NextTrainAvailableInfo` entity from the central ontology. In particular, this mapping establishes an equivalence relation between `vtplat:to` and `vtplat:from` datatype properties, from the source

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<align:Alignment>
 <sripas:steps rdf:parseType="Collection">
      <sripas:step sripas:order="0" sripas:cell="http://www.inter-iot.eu/sripas#cell0"/>
      <sripas:step sripas:order="1" sripas:cell="http://www.inter-iot.eu/sripas#cell1"/>
      <sripas:step sripas:order="2" sripas:cell="http://www.inter-iot.eu/sripas#cell2"/>
   </sripas:steps>
   <align:map>
      <align:Cell rdf:about="http://www.inter-iot.eu/sripas#cell0">
         <align:entity1 rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
            var:CTX a dbo:Train .
            var:CTY a vtplat:TrainService .
            var:CTY vtplat:hasSchedule var:CTP.
            var:CTP vtplat:delay var:delay.
            var:CTX dbo:provides var:CTY .
            var:CTY vtplat:trainNumber var:numtrain .
         </align:entity1>
         <align:entity2 rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
            var:CTX a casaw:NextTrainAvailable ;
               casaw:hasId var:numtrain;
               casaw:delay var:delay.
         </align:entity2>
         <align:relation>=</align:relation>
         <sripas:typings rdf:parseType="Literal">
            <sripas:typing sripas:about="&var;numtrain" sripas:datatype="&xsd;integer"/>
            <sripas:typing sripas:about="&var;delay" sripas:datatype="&xsd;integer"/>
         </sripas:typings>
      </align:Cell>
   </align:map>
      <align:map>
      <align:Cell rdf:about="http://www.inter-iot.eu/sripas#cell1">
         <align:entity1 rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
            var:CTX a dbo:RailwayStation ;
                vtplat:from var:departure;
                vtplat:to var:arrival.
         </align:entity1>
         <align:entity2 rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
            var:CTX a casaw:NextTrainAvailable ;
               casaw:departureFrom var:departure;
               casaw:arrivalTo var:arrival.
         </align:entity2>
         <align:relation>=</align:relation>
         <sripas:typings rdf:parseType="Literal">
             <sripas:typing sripas:about="&var;departure" sripas:datatype="&xsd;string"/>
            <sripas:typing sripas:about="&var;arrival" sripas:datatype="&xsd;string"/>
         </sripas:typings>
      </align:Cell>
   </align:map>
      <align:map>
      <align:Cell rdf:about="http://www.inter-iot.eu/sripas#cell2">
         <align:entity1 rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
            var:CTX a time:DateTimeInterval.
                        var:CTY vtplat:dateTimeInterval var:CTX.
                        var:CTX vtplat:startAt var:startAt;
                 vtplat:endAt var:endAt.
         </align:entity1>
         <align:entity2 rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
            var:CTX a casaw:NextTrainAvailable;
               casaw:startAt var:startAt;
               casaw:endAt var:endAt.
         </align:entity2>
         <align:relation>=</align:relation>
         <sripas:typings rdf:parseType="Literal">
             <sripas:typing sripas:about="&var;startAt" sripas:datatype="&xsd;string"/>
            <sripas:typing sripas:about="&var;endAt" sripas:datatype="&xsd;string"/>
         </sripas:typings>
      </align:Cell>
   </align:map>
 </align:Alignment>
</rdf:RDF>
```

Listing 1. An example of alignment between Viaggiatreno.it and the Central Ontology

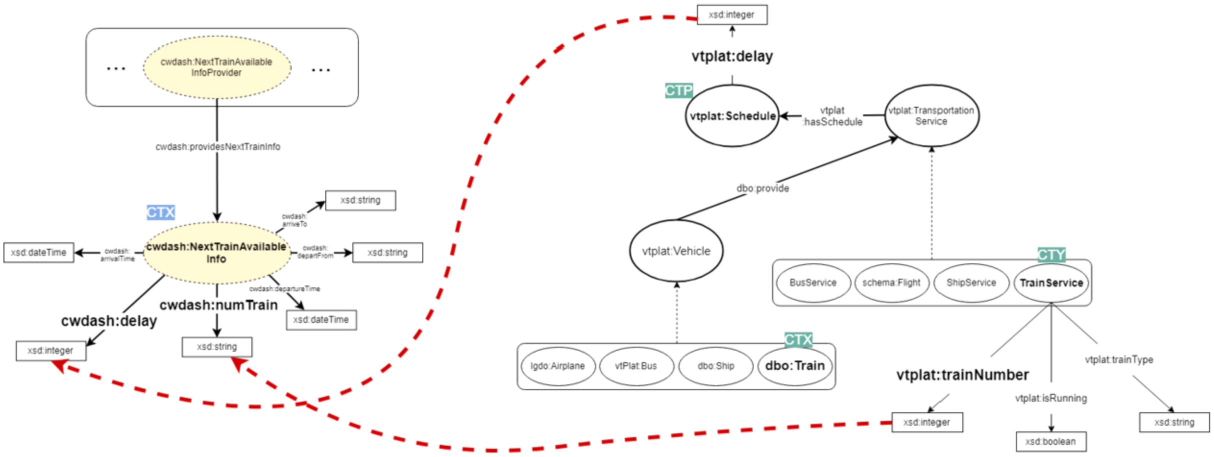

Fig. 11. Alignment fragment example from Viaggiatreno and the Central Ontology.

ontology, and `cwdash:departFrom` and `cwdash:arriveTo`, from the target ontology, respectively (lines from 31 to 49).

## 7. Concluding remarks – discussion of results

The paper presents and discusses the use of INTER-IoT approach to integrate the CasAware platform with other IoT artifacts (platforms and devices). The integration was explained within a domain specific (Ambient Assisted Living) application. To achieve interoperability, appropriate Bridges have been implemented and alignments between data semantics of three IoT platforms created. Prototype implementation has been completed and is being tested across a number of scenarios outlined in [35]. Specifically, the CasAware Dashboard is a GUI-based tool demonstrating the results of the integration of three different IoT platforms within the implementation of the motivating scenario reported in Section 4. In Fig. 12, the activation of the following graphical widgets are shown: a) the external temperature, humidity percentage and precipitation forecast (information received from *Openweather* platform); b) the next available train (information received from *Viaggiatreno* platform; c) the alarm clock with the set time; d) the coffee machine with a time indicator set to the activation time; e) the indoor temperature indicator; f) two alarms alerting the inhabitant, in case of rain and if he/she forgets to take the daily prescribed pill (information handled by the CasAware platform).

This initial proof of concept demonstrated the ease of application of the approach, which indeed has required a quick stage of development and configuration to integrate a new IoT platform. In addition, the evaluation demonstrated that the translation works, as expected, for all connected artifacts. The full communication scenario runs on "conversation specific" alignments, which have allowed to reduce the complexity of the problem of integrating three different platforms and possibly further others (both at application and at semantic level). In this regard, if multiple connected platforms send messages with similar semantics, one alignment can be prepared and reused to consume messages originating at different artifacts.

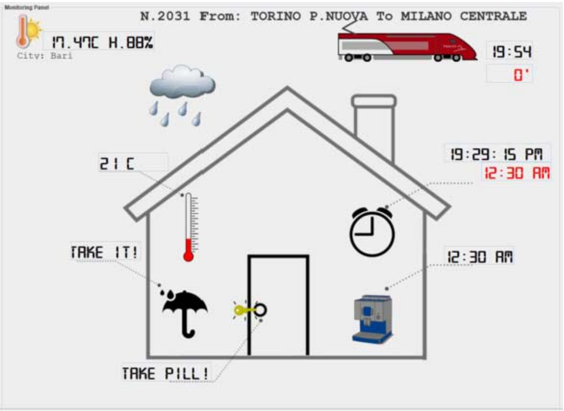

Fig. 12. A screenshot from the CasAware Dashboard.

However, while the tests are successful for a single deployment, one of the interesting open questions is that of scalability. In other words, how will the proposed approach behave in the case of multiple CasAware households being connected to various artifacts (including the two external platforms) via

INTER-IoT middleware/services. Will the current approach scale, and if it does not how it will have to be adapted to obtain sufficient scalability. It is planned to investigate this question in the near future.

Separately, upon reflections based on the initial implementation, the following conclusions are drawn. (1) Instatiation of a Bridge, which connects an artifact to the INTER-IoT environment, is not very difficult and can be achieved by following the templates provided by the INTER-IoT project. (2) Creation of alignments is more complex and requires, at least basic, knowledge about ontologies and semantic technologies. Without such knowledge/understanding what is an ontology, how it is represented, and how it is used in practical applications, creation of correct alignments is a relatively complex and time consuming process, even if appropriate documentation and examples are provided. Here, let us observe that it is a well-known fact that semantic technologies are not very popular, even though they hold a lot of promise. This is, at least in part, because they require knowledge that is not easily available (e.g. few universities include semantic technologies in their CS/CE curriculum). Nevertheless, there must be a way to help alignment developers; for instance, by development of appropriate tools. Here, the additional documentation and alignment patterns,[9] as well as the IPSM-AF editor tool, proposed within the scope of INTER-IoT is a step in the right direction (although the tool was still under development, at the time of writing).

Future developments will also address the efforts toward the two following goals: (i) inclusion of an automatic reasoner in order to extend the knowledge after the semantic alignment process, and (ii) integration of CasAware, leveraging the INTER-IoT approach, of other IoT platforms (e.g. a utility companies platform, an e-commerce platform, an RTLS platform, etc.).

## Acknowledgements

The research reported in this paper has been partially funded by the European Union EU-H2020-ICT under the grant agreement No: 687283, INTER-IoT, and by Regione Lombardia Bando Linea RS per aggregazioni under the grant agreement No: 147152, CasAware.

[9] https://inter-iot-cookbook.readthedocs.io/en/latest/inter-layer/ds2ds/recipes/alignment/

## References

[1] R. Agarwal, D. Gómez Fernández, T. Elsaleh, A. Gyrard, J. Lanza, L. Sánchez, N. Georgantas and V. Issarny, Unified IoT ontology to enable interoperability and federation of testbeds, 2016, pp. 70–75. doi:10.1109/WF-IoT.2016.7845470.

[2] M.I. Ali, N. Ono, M. Kaysar, K. Griffin and A. Mileo, A semantic processing framework for IoT-enabled communication systems, in: *International Semantic Web Conference (2)*, 2015, pp. 241–258.

[3] A. Alliance, Alljoyn framework, *Linux Foundation Collaborative Projects* (2016). https://allseenalliance.org/framework.

[4] D. Amaxilatis, D. Boldt, J. Choque, L. Diez, E. Gandrille, S. Kartakis, G. Mylonas and L. Vestergaard, Advancing experimentation-as-a-service through urban IoT experiments, *IEEE Internet of Things Journal* **PP** (2018), 1. doi:10.1109/JIOT.2018.2871766.

[5] S. Bandyopadhyay and A. Bhattacharyya, Lightweight Internet protocols for web enablement of sensors using constrained gateway devices, in: *Computing, Networking and Communications (ICNC), 2013 International Conference on*, IEEE, 2013, pp. 334–340.

[6] M. Bauer, M. Boussard, N. Bui, F. Carrez et al., Internet of Things – Architecture IoT-A Deliverable D1.5 – Final architectural reference model for the IoT v3.0, 2013.

[7] M. Beigl, H.-W. Gellersen and A. Schmidt, Mediacups: Experience with design and use of computer-augmented everyday artefacts, *Computer Networks* **35**(4) (2001), 401–409. doi:10.1016/S1389-1286(00)00180-8.

[8] O. Bergmann, K.T. Hillmann and S. Gerdes, A CoAP-gateway for smart homes, in: *Computing, Networking and Communications (ICNC), 2012 International Conference on*, IEEE, 2012, pp. 446–450.

[9] M. Blackstock and R. Lea, IoT interoperability: A hub-based approach, in: *Internet of Things (IOT), 2014 International Conference on the*, IEEE, 2014, pp. 79–84. doi:10.1109/IOT.2014.7030119.

[10] D. Brickley and L. Miller, FOAF vocabulary specification 0.99 (2014), Namespace document, 2019, Available online, http://xmlns.com/foaf/spec/.

[11] E.G. Caldarola and A.M. Rinaldi, A multi-strategy approach for ontology reuse through matching and integration techniques, in: *Quality Software Through Reuse and Integration*, Springer, 2016, pp. 63–90.

[12] E.G. Caldarola and A.M. Rinaldi, An approach to ontology integration for ontology reuse, in: *2016 IEEE 17th International Conference on Information Reuse and Integration (IRI)*, IEEE, 2016, pp. 384–393. doi:10.1109/IRI.2016.58.

[13] B. Cheng, S. Longo, F. Cirillo, M. Bauer and E. Kovacs, Building a big data platform for smart cities: Experience and lessons from santander, in: *2015 IEEE International Congress on Big Data*, 2015, pp. 592–599, ISSN 2379-7703. doi:10.1109/BigDataCongress.2015.91.

[14] J. Chin, V. Callaghan and S.B. Allouch, The Internet-of-Things: Reflections on the past, present and future from a user-centered and smart environment perspective, *Journal of Ambient Intelligence and Smart Environments* **11**(1) (2019), 45–69. doi:10.3233/AIS-180506.

[15] M. Compton, P. Barnaghi, L. Bermudez, R. García-Castro, O. Corcho, S. Cox, J. Graybeal, M. Hauswirth, C. Henson, A. Herzog et al., The SSN ontology of the W3C semantic sensor network incubator group, *Web Semantics: Science, Services and Agents on the World Wide Web*, **17** (2012), 25–32.

[16] L. Daniele, F. den Hartog and J. Roes, Created in close interaction with the industry: The smart appliances reference (SAREF) ontology, in: *International Workshop Formal Ontologies Meet Industries*, Springer, 2015, pp. 100–112. doi:10.1007/978-3-319-21545-7_9.

[17] P. Desai, A. Sheth and P. Anantharam, Semantic gateway as a service architecture for iot interoperability, in: *Mobile Services (MS), 2015 IEEE International Conference on*, IEEE, 2015, pp. 313–319. doi:10.1109/MobServ.2015.51.

[18] S. Elias, S. Shivashankar, P. Manoj et al., A REST based design for Web of Things in smart environments, in: *Parallel Distributed and Grid Computing (PDGC), 2012 2nd IEEE International Conference on*, IEEE, 2012, pp. 337–342. doi:10.1109/PDGC.2012.6449842.

[19] ETSI, Smart Appliances REFerence (SAREF) ontology. https://www.etsi.org/technologies/smart-appliances.

[20] G. Fortino, C. Savaglio, C.E. Palau, J.S. de Puga, M. Ganzha, M. Paprzycki, M. Montesinos, A. Liotta and M. Llop, Towards multi-layer interoperability of heterogeneous IoT platforms: The INTER-IoT approach, in: *Integration, Interconnection, and Interoperability of IoT Systems*, R. Gravina, C.E. Palau, M. Manso, A. Liotta and G. Fortino, eds, Springer International Publishing, Cham, 2018, pp. 199–232. doi:10.1007/978-3-319-61300-0_10.

[21] M. Ganzha, M. Paprzycki, W. Pawłowski, P. Szmeja and K. Wasielewska, Towards common vocabulary for IoT ecosystems – preliminary considerations, in: *Intelligent Information and Database Systems, 9th Asian Conference, ACIIDS 2017, Kanazawa, Japan, April 3–5, 2017, Proceedings, Part I*, LNCS, Vol. 10191, Springer, 2017, pp. 35–45. doi:10.1007/978-3-319-54472-4.

[22] M. Ganzha, M. Paprzycki, W. Pawłowski, P. Szmeja and K. Wasielewska, Streaming semantic translations, in: *21st International Conference on System Theory, Control and Computing (ICSTCC), 19–21 Oct. 2017, Sinaia, Romania, Proceedings*, IEEE, 2017, pp. 1–8. doi:10.1109/ICSTCC.2017.8107003.

[23] M. Ganzha, M. Paprzycki, W. Pawłowski, P. Szmeja and K. Wasielewska, Semantic interoperability in the Internet of Things: An overview from the INTER-IoT perspective, *Journal of Network and Computer Applications* **81** (2017), 111–124. doi:10.1016/j.jnca.2016.08.007.

[24] M. Ganzha, M. Paprzycki, W. Pawlowski, P. Szmeja, K. Wasielewska and C.E. Palau, From implicit semantics towards ontologies – practical considerations from the INTER-IoT perspective, in: *14th IEEE Annual Consumer Communications & Networking Conference, CCNC 2017, Las Vegas, NV, USA, January 8–11, 2017*, 2017, pp. 59–64. doi:10.1109/CCNC.2017.7983082.

[25] M. Ganzha, M. Paprzycki, W. Pawłowski, P. Szmeja, K. Wasielewska, B. Solarz-Niesłuchowski and J.S. de Puga García, Towards high throughput semantic translation, in: *Interoperability, Safety and Security in IoT*, G. Fortino, C.E. Palau, A. Guerrieri, N. Cuppens, F. Cuppens, H. Chaouchi and A. Gabillon, eds, Springer International Publishing, 2018, pp. 67–74. ISBN 978-3-319-93797-7. doi:10.1007/978-3-319-93797-7_9.

[26] N. Guarino, D. Oberle and S. Staab, What is an ontology? in: *Handbook on Ontologies*, Springer, 2009, pp. 1–17.

[27] A. Gyrard, C. Bonnet, K. Boudaoud and M. Serrano, Assisting IoT projects and developers in designing interoperable semantic web of things applications, in: *2015 IEEE International Conference on Data Science and Data Intensive Systems*, 2015, pp. 659–666. doi:10.1109/DSDIS.2015.60.

[28] C.A. Henson, J.K. Pschorr, A.P. Sheth and K. Thirunarayan, SemSOS: Semantic sensor observation service, in: *2009 International Symposium on Collaborative Technologies and Systems*, 2009, pp. 44–53. doi:10.1109/CTS.2009.5067461.

[29] I. Horrocks, P.F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, M. Dean et al., SWRL: A semantic web rule language combining OWL and RuleML, *W3C Member submission* **21** (2004), 79.

[30] INTER-IoT Consortium, INTER-MW online documentation, https://inter-iot.readthedocs.io/en/latest/.

[31] G. Kortuem, F. Kawsar, V. Sundramoorthy and D. Fitton, Smart objects as building blocks for the Internet of Things, *IEEE Internet Computing* **14**(1) (2010), 44–51. doi:10.1109/MIC.2009.143.

[32] J.A. Kunze and T. Baker, The Dublin core metadata element set, 2007.

[33] M. Lenzerini, Data integration: A theoretical perspective, in: *Proceedings of the Twenty-First ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, ACM, 2002, pp. 233–246. doi:10.1145/543613.543644.

[34] Linked Open Vocabularies (LOV), https://lov.linkeddata.es/dataset/lov/.

[35] A. Mahroo, D. Spoladore, E.G. Caldarola, G. Modoni and M. Sacco, Enabling the smart home through a semantic-based context-aware system, in: *Second International Workshop on Pervasive Smart Living Spaces, IEEE International Conference on Pervasive Computing and Communication*, IEEE, 2018.

[36] F. Mattern, From smart devices to smart everyday objects, in: *Proceedings of Smart Objects Conference*, 2003, pp. 15–16.

[37] G. Modoni, A. Trombetta, M. Veniero, M. Sacco and D. Mourtzis, An event-driven integrative framework enabling information notification among manufacturing resources, *International Journal of Computer Integrated Manufacturing* **32**(3) (2019), 241–252. doi:10.1080/0951192X.2019.1571232.

[38] G.E. Modoni, E.G. Caldarola, M. Sacco, K. Wasielewska, M. Ganzha, M. Paprzycki, P. Szmeja, W. Pawlowski, C.E. Palau and B. Solarz-Niesłuchowski, Integrating the AAL CasAware platform within an IoT ecosystem, leveraging the INTER-IoT approach, in: *Proceedings of First International Conference on Computing, Communications, and Cyber-Security (IC4S 2019)*, Springer, 2020.

[39] G.E. Modoni, M. Veniero and M. Sacco, Semantic knowledge management and integration services for AAL, in: *Italian Forum of Ambient Assisted Living*, Springer, 2016, pp. 287–299.

[40] G.E. Modoni, M. Veniero, A. Trombetta, M. Sacco and S. Clemente, Semantic based events signaling for AAL systems, *Journal of Ambient Intelligence and Humanized Computing* (2017), 1–15.

[41] OGC, Sensor Web Enablement. http://www.opengeospatial.org/ogc/markets-technologies/swe.

[42] A. Prati, C. Shan and K.I.-K. Wang, Sensors, vision and networks: From video surveillance to activity recognition and health monitoring, *Journal of Ambient Intelligence and Smart Environments* **11**(1) (2019), 5–22.

[43] E. Rahm, The case for holistic data integration, in: *East European Conference on Advances in Databases and Information Systems*, Springer, 2016, pp. 11–27. doi:10.1007/978-3-319-44039-2_2.

[44] F. Ramparany, F.G. Marquez, J. Soriano and T. Elsaleh, Handling smart environment devices, data and services at the semantic level with the FI-WARE core platform, in: *Big Data (Big Data), 2014 IEEE International Conference on*, IEEE, 2014, pp. 14–20. doi:10.1109/BigData.2014.7004417.

[45] P. Saint-Andre, Extensible messaging and presence protocol (XMPP): Core, Technical Report, Internet Engineering Task Force (IETF), 2011.

[46] L. Sánchez, J. Galache, V. Gutierrez, J. Hernández-Muñoz, J. Vercher, A. Gluhak and T. Garcia, SmartSantander: The meeting point between Future Internet research and experimentation and the smart cities, 2011.

[47] A. Sojic, W. Terkaj, G. Contini and M. Sacco, Modularising ontology and designing inference patterns to personalise health condition assessment: The case of obesity, *Journal of biomedical semantics* **7**(1) (2016), 12. doi:10.1186/s13326-016-0049-1.

[48] J. Soldatos, N. Kefalakis, M. Hauswirth, M. Serrano, J.-P. Calbimonte, M. Riahi, K. Aberer, P.P. Jayaraman, A. Zaslavsky, I.P. Žarko et al., Openiot: Open source Internet-of-Things in the cloud, in: *Interoperability and Open-Source Solutions for the Internet of Things*, Springer, 2015, pp. 13–25. doi:10.1007/978-3-319-16546-2_3.

[49] STIIMA-CNR, IoT Living Lab in Lecco. http://www.fhffc.it/progetto/domus-living-lab/.

[50] P. Szmeja, M. Ganzha, M. Paprzycki, W. Pawłowski and K. Wasielewska, Declarative ontology alignment format for semantic translation, in: *2018 3rd International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU)*, 2018, pp. 1–6. doi:10.1109/IoT-SIU.2018.8519921.

[51] M. Uschold, Creating, integrating and maintaining local and global ontologies, in: *Proceedings of the First Workshop on Ontology Learning (OL-2000) in Conjunction with the 14th European Conference on Artificial Intelligence (ECAI-2000)*, Citeseer, 2000.

[52] W3C Consortium, Semantic Sensor Network Ontology, https://www.w3.org/TR/vocab-ssn/.

[53] W3C Consortium, PROV-O, The PROV Ontology, https://www.w3.org/TR/prov-o/.

[54] W3C Consortium, Linked Geo Data, http://linkedgeodata.org/OSM.

[55] W3C Consortium, Basic Geo Vocabulary, https://www.w3.org/2003/01/geo/.

[56] W3C Consortium, RDF. Online. https://www.w3.org/RDF/.

[57] W3C Consortium, OWL. Online. http://www.w3.org/TR/owl2-manchester-syntax/.

[58] D. Wilson, S. McLoughlin and M. Brynskov, Organicity: Lessons from an experimentation as a service model for digital civic innovation, 2019.