Semantic Interoperability on the Web of Things: The Semantic Smart Gateway Framework

Konstantinos Kotis
VTT Technical Research Center
Tampere, Finland
e-mail: Ext-Konstantinos.Kotis@vtt.fi

Artem Katasonov
VTT Technical Research Center
Tampere, Finland
e-mail: Artem.Katasonov@vtt.fi

Abstract—The aim of this paper is to present authors' proposal regarding semantic interoperability for interconnected and semantically coordinated smart entities in a Web of Things. More specific, the paper presents a use case scenario and requirements related to the semantic registration, coordination and retrieval of smart entities. Motivated by these, the paper accentuates the need for, and emphasizes, a framework of Semantic Smart Gateways (SSGF) in the Semantic Web of Things (SWoT), proposing an ontology learning and an ontology alignment method respectively.

Keywords - ontology alignment, semantic coordination, smart entity, smart gateway, semantic interoperability

I. INTRODUCTION

Internet of Things (IoT) is a forthcoming technological revolution that will radically change our environment and enable innovative global as well as local applications and services for users. IoT will enable a global connectivity between physical objects (connecting 'things', not only places or people), will bring real-time machine-published information to the Web, as well as will enable a better interaction of people with the physical environment by combining ubiquitous access with the cloud intelligence.

The 'things' on the IoT are various physical entities that are of some interest to humans, e.g., a heater to control, a package to track, an industrial machine to monitor, an electrical current to measure. Depending on the nature of these things, different technologies for connecting them to the IoT are (or will be) used. The three major options for this come from the three major technology areas related to IoT. They rely on different technologies and are prevailing in different industry sectors, resulting in the first layer of heterogeneity on IoT:

- Attached devices: Identifiers such as RFID tags or barcodes are attached to things to enable their automatic identification and tracking.
- Sensing and Actuating devices: These devices are placed in the close vicinity of the 'things' and provide a "second-hand" access (from outside) to their properties or functions.
- Embedded devices: 'Things' like industrial machinery, home electronics, smart phones, wearable devices have embedded processors, data storages, sensors and actuators, enabling "first hand" access (from inside) to them, often over IP and without specific gateways.

Therefore, IoT will necessarily consist of a heterogeneous set of devices and heterogeneous communication strategies between the devices. It is clear, however, that such a heterogeneous system should evolve into a more structured set of solutions, where 'things' are made uniformly discoverable, enabled to communicate with other entities, and are closely integrated with Internet infrastructure and services, regardless of the particular way (RFIDs, sensors, embedded devices) in which they are connected to the IoT.

The IoT will require interoperability at multiple levels. On the hardware side, such problems have to be addressed as handling a capability mismatch between traditional Internet hosts and small devices, as well as handling widely differing communication and processing capabilities in different devices. In the interface between the device and network domains, IoT gateways will provide a common interface towards many heterogeneous devices and networks. Some IoT devices, e.g. home electronic appliances, will, however, be likely connected directly to the Internet without such middle-boxes.

A trend in IoT area is to attempt to integrate 'things' seamlessly with the existing Web infrastructure and to expose connected 'things' uniformly as Web resources, resulting in what is called the Web of Things (WoT). Such an approach is a great facilitator of interoperability. For true interoperability, we need, however, semantic interoperability, the ability of the devices to unambiguously convey the meaning of data they communicate over Web protocols. Semantic Web (SW) technology can be used to extend WoT into what is sometimes referred to as the Semantic Web of Things (SWoT). Such an extension incorporates into the IoT domain all the benefits of the SW, i.e. a) a web-scale approach using URIs and HTTP, b) extensibility through the Open World Assumption principle, c) interlinking of domain models through inter-model references, d) use of standard languages, and e) model expressiveness through inference of logical consequences.

The work reported in this paper is particularly motivated by a vision of an open and interoperable IoT, where the following four related requirements are satisfied:

- Ability to have gradually growing IoT environments, contrasted to installing and interconnecting all IoT devices and software at once.

This work was carried out during the tenure of an ERCIM "Alain Bensoussan" Fellowship Programme. This Programme is supported by the Marie Curie Co-funding of Regional, National and International Programmes (COFUND) of the European Commission.

- Ability to interconnect devices from different vendors.
- Ability of 3rd parties to develop software applications for IoT environments, contrasted to applications coming only from the devices' vendors.
- Ability to develop applications that are generic in the sense of running on various IoT device sets (different vendors, same purpose), contrasted to developing applications for a very particular configuration of devices.

This vision presumes a rather loosely-coupled interoperability than any other form of tighter integration such as standardization of protocols and application programming interfaces. Therefore, a decision is to rely on the above benefits of the semantic technology. Moreover, we do not assume an agreed and shared ontology that all the device and software vendors would follow at design time, as we find such an assumption unrealistic. Our work proposes however a reference ontology, but the goal of this ontology is to act as a mediator tool when aligning domain ontologies of various vendors and not as a pre-agreement between those vendors (at design-time).

The goal of our work is to support a (semi-)automated centralized (or peer-to-peer under some circumstances) translation process at runtime, with minimum human (end-user) involvement, by computing and storing alignments of data descriptions (i.e. ontological definitions) in such a way that they could be utilized together with the represented data, in a uniform way. This goal extends the work of dynamic ontology linking process for the behavioral coordination of heterogeneous systems that is reported in [1], and complements the related work on ontology mapping in Smart-M3 smart space that is reported in [2]. The proposed work is also aware of and motivated by the related work of semantic coordination of agents in P2P systems [3, 4] in respect to the automatic and self-organization of schema/ontology mappings towards 'healing' incorrect mappings of entities in smart spaces. These related works however are emphasizing on the decentralized computation of mappings, a goal that we will consider in future work. Moreover, our work is motivated by the latest effort within the SWISS-Experiment project, emphasizing the use of W3C XG SSN ontology in a large-scale federated sensor network for semantic data sensor search [6], which manually provides mappings of data to the reference SSN ontology (combined with domain ontologies such as SWEET- NASA ones) using a custom mapping language. Last but not least, in [6], authors present preliminary and on-going work of an ontological framework for the representation and retrieval of connected smart objects in the WoT. The notion of a virtual object and of a compound virtual object is in alignment with the smart entity and smart entity cluster of the work presented in this paper, and the reported process of discovering similarities between connected objects is in alignment with the aim of Smart Gateway framework for semantically coordinating smart entities via their conceptualizations' alignment.

In the IoT setting, we introduce a new high-level type of entities, which we refer to as 'smart entities'. Smart entities 'live' in a domain-specific or cross-domain setting and exhibit some intelligent behavior. They represent physical 'things' and are often equipped with other, connectors in purpose, physical entities (e.g. sensors, but not only). Some smart entities, however, can be virtual and just equipped with datasets/data-streams exposed by Cloud services, e.g. Pachube. Smart entities may also be controlled by other entities responsible for the execution of common tasks, which we refer to as 'control entities'. Furthermore, these smart entities are equipped with a conceptual description of the properties of the physical entities they 'carry' and of the data they produce or consume. This conceptual description, also called an 'ontological definition', is what we aim to provide to entities that are not already equipped with one and to discover similarities between them through a Semantic Smart Gateway's reference ontology.

The paper is structured as follows: Section 2 describes an indicative use case scenario. Section 3 describes the requirements extracted from the scenario and on experience gained from our involvement in Smart Environments and IoT related projects and section 4 the proposed Semantic Smart Gateway design based on these requirements. Section 5 reports on implementation and evaluation plan. Section 6 concludes this paper.

II. USE CASE SCENARIO

In the paragraphs that follow we provide a representative scenario that sketches our requirements for the proposal of the Semantic Smart Gateway Framework (SSGF).

A vendor Vendor-A installs a smart home network HomeSmartNetwork-1 in Mary's house. The network connects two smart entities (SmartE-A1 and SmartE-A2) and a control entity (ControlE-A1). SmartE-A1 is an entity that has a meaning of "smart living room". For simplicity, the smartness here is restricted to being equipped with (equiv. to hasPart property in the IoT ontology) a physical entity which is an electronic device, or more specifically an environmental sensing device for measuring the temperature in the room, i.e., a sensing device equipped with a temperature sensor. SmartE-A2 is a "smart heater", an electric home appliance for heating the house, equipped with an actuating device from remotely adjusting the heating power. ControlE-A1 is a Vendor-A's software controller, running on a computing device, responsible for controlling the interconnection and interoperation of the other two entities. SmartE-A1 is also equipped with a software agent (note that we do not presume any agent architecture here) responsible for a number of tasks related to the particular smart entity, and with an ontology definition that provides semantic description of the schema used for the temperature data, the temperature sensing device itself and to other resources that this entity may be equipped with. Similarly, SmartE-A2 is also equipped with an ontology definition for properties of the device and the language of commands it understands. ControlE-A1 broadcasts to the network a request (query) for temperature data in order to monitor the heat in the living room. SmartE-A1 software agent responds accordingly, returning the current temperature value from the sensor and a timestamp. Based on the preferences of Mary and the control software logic, a related command to increase or decrease the heater power is sent to the heater SmartE-A2 by the ControlE-A1.

A couple of months later, Mary decides to extend the home network with a new smart entity "smart bedroom" also equipped with a temperature sensor. She gets a better offer from Vendor-B, so an order is placed to that vendor. After plugging in the sensor and initiation of a new smart entity SmartE-B1 in the smart home network, a meaning negotiation process is automatically initiated by ControlE-A1 in order to be able to understand the ontology definition of SmartE-B1. The ontology definition of SmartE-B1 is (semi-)automatically aligned by ControlE-A1 with the one of SmartE-A1, which is already known to ControlE-A1. Based on this alignment, ControlE-A1 is now able to request and receive room temperature data from both SmartE-A1 and SmartE-B1 in a unified way, and to control the heater so that the temperature in both rooms is close to Mary's preferences. ControlE-A1 acknowledges the registration of the new smart entity in the domain-specific smart entity cluster it controls.

Later on, Mary learns that Vendor-C provides a generic software application which could be used as an upgrade of the heating control entity for her home. This application, ControlE-C1, is capable of energy-saving predictive heating control that takes into account not only the indoor temperature but also the changes in the outdoor temperature. Mary downloads the ControlE-C1 from a Web-based app store. This is similar to the way in which smartphone applications are downloaded in present. Unlike smartphone applications, however, ControlE-C1 has to discover and align the data provider entities it needs as well as the heater(s). First, it discovers SmartE-A1 and SmartE-B1 present in the house, and aligns own ontology with the ontologies of those to be able to request and receive the indoor temperature measurements. Second, it also discovers the heater SmartE-A2 and aligns with it as well, to be able to send the commands to it. Third, it searches for entities providing the outdoor temperature for the area of Mary's house. As Mary does not have a remotely-accessible outdoor temperature sensor in her network, ControlE-C1 opts for employing SmartE-D1 provided by Vendor-D 'living' in a city smart network, CitySmartNetwork-1 (the logic of Web-based discovery is out of the scope of this paper). SmartE-D1 has the meaning of "smart city environment" and provides the outdoor temperature for Mary's living area. It is not equipped with a sensing device itself but with a Pachube data-feed, which is an approximation based on measurements coming from different outdoor temperature sensors somewhere in the town. Obviously, the ontology of SmartE-D1 has to be aligned with the one of ControlE-C1 as well. In practice, an additional task may have to be first performed which is the automated learning of SmartE-D1 ontology definition based on the Pachube feed metadata (if the ontology definition is not directly available).

TABLE 1. NETWORKING ORGANIZATION INFORMATION OF ENTITIES OF DIFFERENT VENDORS IN THE EXAMPLE SCENARIO

Vendor	Entity	Network organization	
		Pre-	**Post-**
Vendor-A	SmartE-A1	HomeSmartNework-1	WoT-1
	ControlE-A1		
	SmartE-A2		
Vendor-B	SmartE-B1	None	
Vendor-C	ControlE-C1	None	
Vendor-D	SmartE-D1	CitySmartNework-1	

Now, both HomeSmartNetwork-1 and CitySmartNetwork-1 are parts of a particular web of thing instance, namely the WoT-1. This in fact supports the ultimate goal of the presented framework that is, given a set of registered and semantically coordinated smart entities within a WoT, any control entity which is member of the WoT-1 instance (i.e. a registered entity) can run domain/application specific queries for the retrieval of these smart entities and of their properties, and of course of their data, in a unified way. The scenario presented handles only two control and a few physical/smart entities. More elaborated and complex use cases would accentuate more the need for the proposed framework.

III. REQUIREMENTS FOR DESIGNING SEMANTIC SMART GATEWAYS

Based on the above use case scenario and on experience gained from our involvement in smart environments and IoT related projects, we discuss a number of requirements related to the design of semantic smart gateways in the IoT. A Semantic Smart Gateway (IoT-SG) must provide:

a) A semantic way for registering smart entities (their ontological definitions and their alignments also) in a medium-to-small scale, so their discovery and retrieval can be performed in an intelligent manner (using common views of data)

b) An on-the-fly ontology learning component to support the (semi-)automated semantic description of smart entities' that are not pre-equipped with ontological definitions but instead they 'carry' simple metadata e.g. an RDBS schema metadata as in Pachube cloud service

c) An on-the-fly ontology alignment component to support the run-time discovery of similarities between smart entities' ontological definitions (and between smart entities themself respectively) through a shared reference ontology, in order to be able to i) support the retrieval of similar smart entities in a common view, ii) support the clustering of smart entities into domain-specific clusters, iii) to support the merging of similar smart entities towards a more efficient network organization

d) A semantic retrieval component for allowing agents (software or humans) to place ontology-driven queries for discovering smart entities that match specific configurations or properties. This component however may not be considered a major part of the Semantic Smart Gateway since it can be engineered at any time, independently, by utilizing a SPARQL end-point reference.

Agents must be able to automatically register new smart entities or retrieve already registered ones that match certain properties/criteria. Newly added heterogeneous smart entities, carrying their vendors' ontological descriptions (or other types of metadata) for describing properties of the devices or data they 'carry', must be coordinated by automatically aligning their ontological definitions with the definitions of other smart entities, in a direct (point-to-point) way or via the shared semantics of a reference ontology (i.e. the IoT-ontology) for more precise and fast computations.

In a Semantic Smart Gateway framework approach, due to the 'on-the-fly' requirement of performing computations with smart entities (at runtime), we conjecture that precision and speed of ontology alignment computations is of major importance, thus the choice of a centralized reference point for computing similarities of peer points was made. So the time needed only to compute the similarity of an ontological definition (with a translation need) of a newly registered smart entity against the reference ontology is exponentially less than the time to perform the same computation against each peer smart entity registered in a Semantic Smart Gateway (given that the size of reference ontology and peer ontologies is almost the same). Moreover, during the last 6 years of experience in the ontology alignment (mapping) evaluation initiative [7] and its contest, we have observed that ontology alignment methods which utilize reference ontologies produce more precise results than others. Although to depend on such a reference ontology generally is not a very good practice (must ensure that one exists or engineer one that meets your requirements) however in our setting seems to have more benefits than obstacles.

IV. THE SEMANTIC SMART GATEWAY DESIGN

Registered smart entities should be semantically coordinated towards a) retrieving their data or other information required by intelligent applications in a unified way, b) automatically shaping clusters of domain-specific or cross-domain entities, and c) merging similar smart entities for efficiency reasons. Towards this coordination, the task of the on-the-fly (semi-)automated alignment of smart entities' ontological definitions (at run-time) is of major importance thus the design decision to integrate it in the SGF was made, supporting it by the incorporation of related information in the IoT-ontology. Such information is based on to the ODP ontology alignment design patterns. It must be stated that the IoT-SGF and the IoT-ontology support a hybrid architecture, allowing both distributed and centralized topologies of either autonomous and intelligent smart entities (first case) or of entities that rely on the shared vocabulary of the IoT-ontology (second case). However, in this preliminary work we will emphasize the centralized topology for the aforementioned reasons in section III.

Regarding automation in terms of human-involvement, the proposed framework considers the amount and type of human involvement that is really needed in order to a) learn an ontology from metadata as much close to the intended one as possible, and b) to compute ontology alignments as precise and fast as possible. Human agents in the involved tasks are non-experts in knowledge engineering but trained to perform installation/upgrade of smart entities in smart networks and use the installation/upgrade support software of a Semantic Smart Gateway. These agents will have to evaluate at runtime the ontology learned from metadata attached to a Smart Entity that is newly added in the network and also to evaluate the computed alignments between newly added smart entities and the IoT-ontology. Based on their evaluation end-user must be provided with the necessary tools to edit and fix them appropriately.

A. The semantic registry (IoT-ontology)

Due to space limitations, in this paper we do not present or discuss the IoT ontology (IoT-ontology) for supporting the IoT-SGF functionality. This is a parallel on-going process under continuous development and pending publication. A latest experimental and populated version of this ontology, together with the necessary imported vocabularies, can be found at http://purl.org/IoT/iot. This ontology represents different types of data, metadata, resources and devices that are parts of smart entities. Also, the ontology represents metadata for description of alignments between different ontological definitions that each smart entity is also equipped with. This representational effort is related only to the extent of a Semantic Smart Gateway and to the description of things that enable humans and machines to understand what services they provide, enabling their search and automated discovery in an architecture for an open WWW of Things, as it is reported in Guinard et al, 2010 [8].

The ontology learning task as well as the task of computing alignments between ontological definitions of smart entities is performed using the IoT-ontology as a reference ontology. The resulted alignment metadata are stored in the IoT-ontology as ontology alignment instances.

B. On-the-fly ontology learning

To support the abovementioned scenario and requirements of a Semantic Smart Gateway, an on-the-fly ontology learning component must be designed to support the (semi-)automated semantic description of smart entities that are not pre-equipped with ontological definitions. We conjecture that the most common case in this setting is to have entities that describe their data using simple metadata such as an RDBS schema. A real-case example is the Pachube Cloud service metadata for describing sensor data where temperature sensed data stream/feed is described with the properties ID (e.g. 'temperature'), Units (e.g. 'Celsius'), and with the following metadata: Location Name, Location Map, Latitude, Longitude, Elevation, Exposure, Disposition, Domain, Feed Format, Website, Contact Email, and DataStream. The aim of the proposed functionality is to automatically or at least semi-automatically generate a lightweight temperature domain ontology using this information at runtime.

An ontology structure can simply defined as $O_1 = (S_1, A_1)$, where S denotes the signature (set of concepts) and A the set of axioms that specify the intended meaning of terms in S. Based on the ontological structure, ontology comprises a set of instances, which could be seen as the extension of concepts. This paper adopts the latest standard recommended by W3C, OWL 2.0, as the ontology description language.

Object-Relational database systems are based on the Object-Relational model, extending the relational data model by providing richer types. Due to Object-Relational database system's features, it has been considered a favorite resource for the ontology learning since it is easier to obtain the ontology concepts than doing that by leaning them from RDB. For these reasons we will reuse (Object-)Relation-to-OWL learning rules as these are reported in [9] and [10].

The proposed learning strategy is outlined in the following steps:

a) Automatically compute (Object)Relation-to-OWL mappings on-the-fly. By 'default' here we mean simple mappings of RDB schema to OWL ontology elements following specific learning rules. In case of a different input notation, e.g. XML or JSON, an additional convertor to OWL specification will be utilized.

b) Allow agents (in the simplest case, these will be designers) to inspect and modify names of classes/properties where necessary, in order to modify the learned definitions, providing a NaturalOWL-like or ACE-like control language and graph-like representation of the learned ontology, as an interface for editing the initial learned ontological definitions in a natural way, hiding symbolic language from agents [11].

C. On-the-fly ontology alignment

To further support the scenario and requirements of a Semantic Smart Gateway as presented in this paper, an on-the-fly ontology alignment component must be designed to support the (semi-)automated discovery of similarities between smart entities that are equipped with ontological definitions.

The problem of computing alignments between ontologies can be formally described as follows: Given two ontologies $O_1 = (S_1, A_1)$, $O_2 = (S_2, A_2)$ (where S_i denotes the signature and A_i the set of axioms that specify the intended meaning of terms in S_i) and an element (class or property) E_i^1 in the signature S_1 of O_1, locate a corresponding element E_j^2 in S_2, such that a mapping relation (E_i^1, E_j^2, r) holds between them. r can be any relation such as the equivalence ($\equiv$) or the subsumption ($\sqsubseteq$) axiom or any other object property e.g. 'partOf'. For any such correspondence a mapping method may relate a value γ that represents the preference to relating E_i^1 with E_j^2 via r. If there is not such a preference, we assume that the method equally prefers any such assessed relation for the element E_1. The correspondence is denoted by $(E_i^1, E_j^2, r, \gamma)$. The set of computed mapping relations produces the mapping function $f: S_1 \rightarrow S_2$ that must preserve the semantics of representation: i.e. all models of axioms A_2 must be models of the translated A_1 axioms: i.e. $A_2 \models f(A_1)$.

The synthesis of alignment methods that exploit different types of information and that compute different types of relations between elements (e.g. equivalence or subsumption relations) has been already proved to be of great benefit [4, 12, 13]. In this preliminary work we follow the simple synthesis strategy which performs composition of results: the results of individual methods are combined using specific operators, e.g. by taking the union or intersection of results, intersection of results or by combining the methods' different confidence values with weighing schemas. Given a set of K alignment methods (e.g. string-based, vector-based), each method has its own confidence values concerning any assessed relation (E_1, E_2, r). The synthesis of these K methods aims to compute an alignment of the input ontologies, with respect to the confidence values of the individual methods. Trimming of the resulted correspondences in terms of a threshold confidence value is then performed for optimization.

The proposed mapping strategy is outlined in the following steps:

Step 1: For each integrated alignment method K compute correspondence $(E_i^1, E_j^2, r, \gamma)$ between elements of a peer's domain ontology and a reference domain ontology definition that is integrated in the IoT-ontology.

Step 2: Apply synthesis of methods (using different aggregation operators) to their resulted sets S_k

Step 3: Apply trimming process by allowing agents to change a variable threshold value for each result set S_k or for the result of a synthesized method.

Step 4: Visualize alignments [14] for inspection by human agents and allow for modifications either by directly changing the alignments on a graph-like representation of the alignments or by allowing the modification of threshold and synthesis operator values in order to re-run alignment methods.

The proposed ontology alignment approach considers most of the challenges in OM research but emphasizing the a) alignment method selection, synthesis and tuning, and b) user involvement, as these where initially reported in [15] and in [16].

V. IMPLEMENTATION AND EVALUATAION PLAN

To implement the presented proposed SSGF we plan to reuse and integrate existing open source or state-of-the-art methods and APIs (e.g. Ontology Alignment API [19]) in every step of the development processes. It is expected that experience gained from past research on both ontology learning [17] and alignment [12, 13, 15] research lines as well as on smart semantic middleware for the Internet of Things [18] will be a major contribution in both the development and the evaluation phase of this framework.

After the evaluation of a Semantic Smart Gateway prototype, the extension of reused methods or the development of new ones will be considered to overcome possible issues in the precision and speed of the ontological definitions' alignment computations as well as in the efficiency of the ontology learning functionality.

Regarding evaluation data, we plan to evaluate the ontology alignment functionality using a relative subset of OAEI contest ontologies as well as other experimental ontologies that will be developed from scratch due to need to represent knowledge with particular characteristics: very few concepts (one concept in most cases), sallow hierarchy if any, few or no properties, mainly data properties instead of object properties, representation of task or command related knowledge (verbs).

Since this accepted version of the paper, authors have already implemented and start evaluating a) the IoT-ontology as part of the semantic registry and b) an ontology alignment tool that one of its versions has been submitted for evaluation in the Ontology Alignment Evaluation Initiative (OAEI 2011.5) contest.

VI. CONCLUSSION

This paper reports on a Semantic Smart Gateway Framework proposal towards supporting semantic interoperability of 'things' in the IoT. The framework is designed to support the on-the-fly semi-automated translation process of Smart Entities' data at runtime, with minimum human (designer and end-user) involvement, by computing their ontological definitions' alignments via a mediated schema and storing them in such a way that they could be queried and utilized together with other data in a uniform way.

ACKNOWLEDGMENT

Authors acknowledge that part of the work presented in this paper has been motivated by initial requirements that have been partially specified in the following projects: IoT-SRA (Finnish), USEFIL and iCORE (FP-7).

REFERENCES

[1] A. Katasonov and V. Terziyan, Using Semantic Technology to Enable Behavioral Coordination of Heterogeneous Systems, In: Semantic Web, 2010, pp. 135-156.

[2] A. Smirnov, A. Kashevnik, S. Shilov et al, On-the-Fly Ontology Matching in Smart Spaces: A Multi-model Approach, In: S. Balandin et al. (Eds.): ruSMART/NEW2AN 2010, LNCS 6294, pp. 72–83, 2010

[3] P. Cudré-Mauroux, A. Suchit, and A. Karl, GridVine: An Infrastructure for Peer Information Management and Large-Scale Collaboration, IEEE Internet Computing, 11(5), pp. 36-44, 2007.

[4] V. Spiliopoulos, G. A. Vouros, Synthesizing Ontology Alignment Methods Using the Max-Sum Algorithm, Knowledge and Data Engineering, IEEE Transactions on, vol.PP, no.99, pp.1-1.

[5] J. Calbimonte, H. Jeung, O. Corcho and K. Aberer, Semantic Sensor Data Search in a Large-Scale Federated Sensor Network, In: Semantic Sensor Network Workshop, ISWC 2011, Bonn, 2011

[6] C. Benoit, V. Verdot, V. Toubiana, Searching the 'Web of Things', In: Fifth IEEE International Conference on Semantic Computing (ICSC), 2011, pp. 308 – 315

[7] J. Euzenat, C. Meilicke, H. Stuckenschmidt, P. Shvaiko, C. Trojahn: Ontology Alignment Evaluation Initiative: six years of experience Journal on Data Semantics, 2011

[8] D. Guinard, V. Trifa, E. Wilde, Architecting a Mashable Open World Wide Web of Things, Technical Report No. 663, Department of Computer Science, ETH Zurich, February 2010

[9] M. Li, X. Du, S. Wang, Learning ontology from relational database, In: Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou, 18-21 August 2005, pp. 3410 – 3415, Vol. 6

[10] J. Chen, Y. Wu, M. Li, S. Li, J. You, Rules Driven Object-Relational Databases Ontology Learning, In: Proceedings of International Conference on Interoperability for Enterprise Software and Applications China, 2009, pp. 169-174

[11] K. Kotis, A. Papasalouros, G. A. Vouros, N. Pappas, and K. Zoumpatianos, Enhancing the Collective Knowledge for the Engineering of Ontologies in Open and Socially Constructed Learning Spaces, Journal of Universal Computer Science, vol. 17, issue 12, pp. 1710--1742, 08/2011

[12] Kotis, K., A. Valarakos, and G. Vouros, "AUTOMS: Automating Ontology Mapping through Synthesis of Methods.",International Semantic Web Conference (ISWC'06), Ontology Matching International Workshop, Atlanta USA, 00/2006.

[13] Valarakos, A., V. Spiliopoulos, K. Kotis, and G. Vouros, "AUTOMS-F: A Java Framework for Synthesizing Ontology Mapping Methods", i-Know,07, Graz, Austria, 00/2007

[14] Monika Lanzenberger, Jennifer Sampson: Human-Mediated Visual Ontology Alignment. HCI (9) 2007: 394-403

[15] Konstantinos Kotis, Monika Lanzenberger: Ontology Matching: Current Status, Dilemmas and Future Challenges. CISIS 2008: 924-927

[16] Pavel Shvaiko, Jérôme Euzenat: Ten Challenges for Ontology Matching. OTM Conferences (2) 2008: 1164-1182

[17] Kotis, K., and A. Papasalouros, "Learning useful kick-off ontologies from Query Logs: HCOME revised", 4th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS-2010), Kracow, IEEE Computer Society Press, 2010

[18] A. Katasonov, O. Kaykova, O. Khriyenko, S. Nikitin and V. Terziyan, Smart semantic middleware for the Internet of Things, In: Fifth International Conference on Informatics in Control, Automation and Robotics, Vol. ICSO, 2008, pp. 169–17

[19] J. David, J. Euzenat, F. Scharffe, C. Trojahn dos Santos, The Alignment API 4.0, Semantic Web - Interoperability, Usability, Applicability, 2(1):3-10, 2011, IOS Press