

# Capturing the contributions of the semantic web to the IoT: a unifying vision

<sup>a</sup> *IRIT, UMR 5505, UT2J, Département de Mathématiques-Informatique, 5 allées Antonio Machado, F-31058 Toulouse Cedex, France*

*E-mail: firstname.lastname@irit.fr*

<sup>b</sup> *LAAS-CNRS,*

*7 avenue du Colonel Roche,*

*F-31400 Toulouse, France*

*E-mail: firstname.lastname@laas.fr*

<sup>c</sup> *Univ de Toulouse, INSA, LAAS, F-31400, Toulouse, France*

Nicolas Seydoux <sup>a,b,c</sup>, Khalil Drira <sup>b,c</sup> Nathalie Hernandez <sup>a</sup> Thierry Monteil <sup>b,c</sup>

**Abstract.** The Internet of Things (IoT) is a technological topic with a very important societal impact. IoT application domains are various and include: smart cities, precision farming, smart factories, and smart buildings. The diversity of these application domains is the source of the very high technological heterogeneity in the IoT, leading to interoperability issues. The semantic web principles and technologies are more and more adopted as a solution to these interoperability issues, leading to the emergence of a new domain, the Semantic Web Of Things (SWoT). Scientific contributions to the SWoT are many, and the diversity of architectures in which they are expressed complicates comparison. To unify the presented architectures, we propose an architectural pattern, LMU-N. LMU-N provides a reading grid used to classify processes to which the SWoT community contributes, and to describe how the semantic web impacts the IoT. Then, the evolutions of the semantic web to adapt to the IoT constraints are described as well, in order to give a twofold view of the convergence between the IoT and the semantic web toward the SWoT.

**Keywords:** Semantic Web of Things, Architectural pattern, Internet of Things, Semantic web, survey

## 1. Introduction

The internet has evolved toward a **globally connected environment**. The interconnection of devices, smartphones, services, etc, form what is called the **Internet of Things (IoT)**. A detailed definition of the IoT is provided in [1], embracing the diversity of nature and purpose of the so-called Things. In this paper, the term IoT refers to the technologies and principles enabling the deployment of devices and services networks. IoT networks are based on very diverse technological stacks, for hardware, software and communications. The integration of traditional technologies of the web in the IoT, such as URIs for resource naming, HTTP connectivity, or REST interfaces, is called the **Web Of Things (WoT)**, defined by [2] as "A way to realize the IoT where (physical and virtual) things

are connected and controlled through the World Wide Web". This definition is used as a reference by the W3C Web of Things Working Group<sup>1</sup>.

Indeed, the IoT and WoT "Things" are now an every day life reality for many people. To support this claim, one might consider the increasing number of smart cities including but not limited to Dublin (IR)<sup>2</sup>, Santander (ES)<sup>3</sup>, Milton Keynes (UK), San Francisco (US), New York (US), Yokohama (JA), that are being equipped in order to monitor their environment and to offer innovative services. It is estimated that by 2020, **26 billion devices** will be connected by Machine-to-Machine (M2M) technologies, compared to the 0.9 bil-

---

<sup>1</sup><https://www.w3.org/WoT/WG/>

<sup>2</sup><http://smartdublin.ie/>

<sup>3</sup><http://www.smartsantander.eu/>

lion connected in 2009[3]. The exponential multiplication of connected devices is correlated with a dramatic increase in the volume of exchanged data.

The growing integration of connected Things to human activities is partly driven by the wide scope of application domains covered by the IoT: environmental metering, transportation, home automation, e-health, agriculture, manufacturing...

However, the diversity of IoT-based applications is restricted by the approach of the industry so far, which has been oriented toward **vertical silos**, as [4] points out. Proprietary systems are designed with a specific purpose, and on top of the devices the vendor also distributes the application serving the purpose. This approach cannot match the diversity of possible scenarios driven by user requirements, and raises **interoperability issues**. Users should be allowed to combine their connected devices in a personalized fashion, and an application developers should be able to deploy generic applications that adapt to the available devices, which is totally opposed to vertical integration. Furthermore, such customization of one's IoT network requires devices to be able to understand each other: they must be **semantically interoperable**. [5] and [6] make a distinction between syntactic interoperability, or the ability for systems to exchange content, and semantic interoperability, the ability to understand the exchanged messages. The extreme technological diversity[7], as well as the multiplicity of standards (including but not limited to oneM2M<sup>4</sup>, OIC<sup>5</sup>, AllJoyn<sup>6</sup>, LWM2M<sup>7</sup>), makes syntactic interoperability at a large scale a non-trivial issue. However, solving syntactic interoperability issues only is not sufficient to achieve the complete potential of the IoT, and it is not in the scope of this paper.

Semantic interoperability is an important enabler for the future development of the IoT [8]. Indeed, the core of the IoT is M2M communication. For instance, the observations gathered by sensors are meant to be distributed to other devices, and not to be human readable without processing. Since no human can re-contextualize exchanges between devices, **messages should be understandable from a machine to another**.

As [9] or [8] points out, semantic web principles and technologies can provide solutions to the issues the

IoT is facing. The use of dereferencable vocabularies such as ontologies enables the capture of metadata in a machine-understandable way, easing M2M communication. Many recent research contributions from the semantic web community inject semantic web capabilities (rich content description, reasoning...) into the WoT in order to develop the so-called Semantic Web Of Things (SWoT), an evolution of the WoT where the IoT is extended with semantic web principles and technologies. The publication of resources from an IoT network to the SWoT can feed the Linked Open Data (LOD), in order to make the collected data available.

This paper depicts a landscape of the SWoT, and provides a reading grid to classify the research contributions transforming the IoT into the SWoT. First, to structure the analysis, section 2 defines Lower, Middle and Upper Node (LMU-N), a node-centric IoT architectural pattern. LMU-N provides a framework to support the identification of recurrent patterns in existing research. It is used to describe how the semantic web principles and technologies contribute to the IoT in section 3. On the other hand, section 4 focuses the challenges the semantic web faces to be compliant with the IoT constraints, and how recent semantic web contributions propose to adapt to them. Finally, section 5 concludes this paper and proposes some perspectives for the future of SWoT.

## 2. Unifying the heterogeneity of architectures with LMU-N

After having motivated the need for a unifying architectural pattern, this section presents LMU-N and its main components: nodes and messages flows. Other surveys are listed as related work, before showing how LMU-N can be used as a reading grid for our survey.

### 2.1. Motivations

The papers associating the semantic web and the IoT are many, and their publication rate is increasing: from 15 publications in 2003, to 494 publications in 2016<sup>8</sup>. The total count of publications amounts to 1426. We focused on 71 scientific publications for the survey, chosen for their quality, their innovative aspect, and

<sup>4</sup><http://www.onem2m.org/>

<sup>5</sup><https://openconnectivity.org/>

<sup>6</sup><https://allseenalliance.org>

<sup>7</sup><http://www.openmobilealliance.org>

<sup>8</sup>After a study on <http://ieeexplore.ieee.org>, <http://www.sciencedirect.com> and <http://dl.acm.org>, searching for the exact keywords "semantic web" and "internet of things"

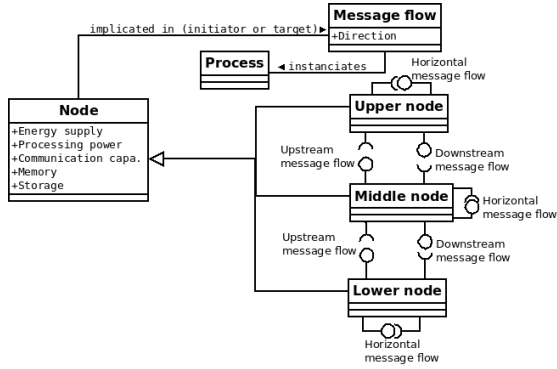


Fig. 1. The LMU-N pattern

for the balance in their content between semantic web and IoT. An extra attention was given to publications proposing semantic web contributions in explicitly and precisely defined IoT architectures. These publications describe contributions that can be competing or complementary, mutually exclusive or not, etc. In order to be able to capture a structured landscape of the contributions of the semantic web to the IoT, one needs to be able to compare these contributions, and to organize them with respect to one another. To do so, we propose **LMU-N, a unifying architectural pattern** that aims at identifying the core components of an IoT network in a generic manner, and to characterize their relationship. LMU-N is then used as a framework to contextualize semantic web contributions to the IoT.

## 2.2. Constituting elements of LMU-N

LMU-N has two major components: the **nodes**, or the "Things" communicating on the IoT, and the **flows**, representing the communications between the nodes. LMU-N is the result of a bottom-up analysis, and its description is linked to a set of pre-existing architectures from which it was deduced. The pattern is represented as an UML model in figure 1. The pattern is composed of three main classes: the **Node**, specialized in three subclasses, the **Message flow**, and the **Process**. The nature of these classes are respectively described in sections 2.2.1, 2.2.2 and 2.3.

### 2.2.1. Representing physical device and virtual service as Nodes

**Definition of an IoT node** The IoT is a network of Things connected together, and can be seen as a **graph where the Things are vertexes and their connections are the edges**. So far, the notion of Thing has been used to indifferently refer to devices (sensors and

actuators) and to services. This choice is justified by the similarity of the roles devices and services have on IoT networks: a service can be seen as the endpoint of a device for the other nodes of the IoT, and a device can be seen as the physical implementation of a service, depending on the chosen perspective.

That is why we introduce in this section the **node**, an abstraction covering both device and service concepts. Fundamentally, **a node is an active entity that can be addressed on the network**. An active entity is able to send and/or receive requests, which is the case for both devices and services. This notion of node is already present in architectures described in [10] or in [11]. [10] describes a contribution based on oneM2M<sup>9</sup>, a standard architecture structured around different nodes which interfaces are formally defined to enforce interoperability. Blurring the line between physical object and virtual service is also at the core of propositions such as [12], [13] or [14], where "virtual entity" is used as an abstraction to enable the composition of services or devices.

Section 3 lists approaches that are either directed to devices or services, and their seamless integration and comparison is enabled by the adoption of the node as the central architectural element. Moreover, presenting nodes over devices or services serves the purpose of interoperability by easing the homogeneous modelling of their capabilities. For instance, virtual entities in [15] are used to capture the characteristics of a node making it relevant to its neighbours.

**Abstracting devices and services** into a generic node also aims at **focusing on their intrinsic characteristics**, such as processing power or communication abilities. Since the early studies on semantic sensor networks, the description of devices and services has been a primary concern of the SWoT community. Contributions like [16], [17] or [18] propose ontologies to define both the capabilities of a device, and to add metadata to the content it handles. Section 2.3.2 details how node descriptions are used to integrate an IoT deployment in the SWoT.

**Classification of LMU-N nodes** **All nodes of an IoT are not equivalent**: some devices are very constrained, whereas the servers providing analytic capabilities are powerful machines. For instance, in its vision of urban IoT, [19] uses the notion of node, and proposes to use two different protocol stacks according to the capabilities of each node (constrained or not).

<sup>9</sup><http://onem2m.org/>

In this paper, different characteristics are used to cluster nodes into meaningful classes constituting LMU-N, represented as attributes of the Node class in fig. 1. The core characteristic of a node is its **processing power**, i.e. its ability to apply treatments of varying complexity to content. The processing power also determines the ability of the node to process content of a varying expressivity, from the very simple agreed-upon byte array to the much more complex Knowledge base (KB) instantiations. The higher a node's processing power is, the more expressive content it can process, and the more complex operations it can achieve. Nodes are also characterized by their **memory**, i.e. the quantity of information it can hold at a given time, and **storage** capability. Storage is the available space giving access to persistent content. The notion of IoT is inseparable from the notion of connectivity, and a node can also be classified according to its **communication capabilities**. These capabilities include the protocols it supports, its general availability on the network, and its bandwidth. Nodes also differ by the nature of their energy source: while some nodes are attached to traditional power grids, other nodes, deployed in the field, are reliant on batteries, or on renewable energy sources like solar panel, or energy harvesting.

Three homogeneous types of nodes are identified, constituting the **three layers** of LMU-N. These subclasses of Node are represented on fig. 1, and are described in tables 2 and 1 regarding their typical characteristics and roles on the network.

- **Upper Node (UN)** are the "weakly constrained nodes" of [19], or Infrastructure Node in [10]. In this category, we classify cloud or local servers as well as powerful mobile devices, including standard laptops and smartphones and domain-specific mobile robots or machines. They have high processing power, extended communication capabilities, and large storage capabilities. For instance, in [20], UN are in charge of applying data fusion operators to data streams. In [21] or [1], the UN are situated in the cloud.
- **Middle Node (MN)** are very often referred to in the literature as **gateway** ([22], [10], [4]), because they are bridges between powerful UN and more constrained nodes. They are usually dedicated to content transformation and protocol bridging: in [23], MN are presented as intelligent nodes where content can be converted from its raw representation to a richer one. [4] proposes an architecture where **the gateway is a contact point between**

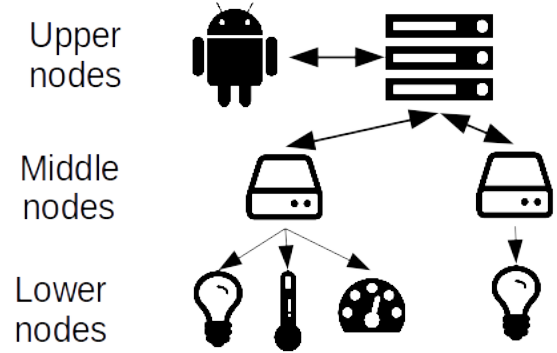


Fig. 2. An instantiation of LMU-N

**the IoT and the SWoT**, performing both annotation and protocol proxying. In [24] and [19], MN are proxies for wireless devices networks. MN typically have medium processing power, extended communication capabilities, and restricted memory storage: they are nodes where content collected by sensors or sensing services is collected, transformed, and redistributed.

- **Lower Node (LN)** are typically connected devices, with very limited power source, processing and communication capabilities, and very limited to no storage capabilities. These are by definition present in every IoT architecture, and in direct contact with the physical world. In some papers such as [20] or [25], these nodes are not directly present, their representation is wrapped by a MN. The LN were in early studies mainly sensors, like in [22], and evolved toward diverse nodes including actuators, displays and composite devices in more recent work such as [10].

Figure 2 represents an instantiation of the LMU-N pattern based on the work described in [18]. Two lamps, a temperature sensor, and a pressure sensors instantiate lower nodes. Their data is collected in a request-response fashion by two gateways. The communication is initiated by the gateway, therefore the arrows are directed from the gateways to the devices. The two upper nodes represent the server hosting the enrichment application, and the robot which processes the content generated by the sensors. The gateways themselves have push capabilities, and can also be queried directly by the server: the communication goes both ways. SmartSantander<sup>10</sup> gives another simple ex-

<sup>10</sup><http://www.smartsantander.eu/>

ample of real-world deployment conform to the LMU-N pattern.

All the architectures presented in this section can be described by LMU-N, supporting its unifying status. A notable exception to the three-layers architecture model is presented in [26], where the authors promote a direct connection between devices and application servers. They propose to this end an architectural approach where each node is equipped with a high-efficiency RDF serializer and tuple store. However, in our opinion, this approach requires the devices to be able to run such a stack, and reduces the possibility to include non semantic-aware, legacy devices in an IoT system. Moreover, we think that directly connecting LN to UN in a flat architecture is a source of scalability issues in large systems. That is why LMU-N is structured with an intermediary level to mediate between potentially very constrained nodes and the rest of the network.

### 2.2.2. Representing exchanges between nodes as messages flows

In LMU-N's representation of an IoT network as a graph, the edges are communications between nodes, instantiated by messages flows. We will refer to these messages' expressivity according to the classification provided by [27] as data, information or knowledge, organized in the Data, Information, Knowledge and Wisdom (DIKW) hierarchy. Typically, **sensors produce data**, and this data is later on contextualized to become information, and enriched using semantic web principles and logic in order to produce knowledge.

Furthermore, the IoT relies on already existing internet technologies and paradigms. Nodes hosting services are servers, and they offer their services to applications or other nodes that are clients in a classic client-server architecture. This model implies asymmetry in the communications among nodes: in a client-server model, the interaction is initiated by the client. This entails that on the graph of nodes, **the edges are oriented**, from the client to the server node.

As shown on fig. 1, flows can be directed in three general directions: **horizontal** for nodes of the same level, such as the work done in [28], **upstream** (as in [29] or [30]), when the source node is of a lower level than its destination node, and **downstream** (as in [31]) otherwise. The notions of upstream and downstream communication in IoT architectures are presented by [31]. The content exchanged in these flows can either be application-dedicated, for instance in [20], [30] or [31], specific to the function of the devices that col-

lected it (temperature observations, user requests...), or system-dedicated, describing the nodes constituting the network, like in [24].

### 2.3. Using LMU-N to classify SWoT contributions

By describing IoT systems with the LMU-N pattern, **different recurring processes are identified, agnostic to the underlying technology stack and to the application domain**. The aim of this paper is to identify and describe the processes that can benefit from an integration into the SWoT, and how they are instantiated in research contributions.

The processes are instantiated by exchanges of messages between nodes, supported by flows as described in section 2.2.2. Therefore, our analysis is based on the assumption that **(i) processes are directed** (horizontal, upstream or downstream) and **(ii) dependant on the characteristics of the nodes at stake**. Contributions brought to the SWoT are dedicated to certain processes, and they can be located in LMU-N depending on the nodes they involve. For instance in [32] or [33], the process of content enrichment is an upstream process, as the content is produced in lower nodes and enriched later on in middle or upper nodes. Processes can also be associated with specific flows content: for instance, in [34], the node exposition process is based on node metadata, and not with applicative content.

All the processes introduced in this paper are identified from published papers after their projection on LMU-N. Two categories of processes are distinguished, presented in the remainder of this section.

#### 2.3.1. Content-related processes

Content-related processes are the activities that are specific to an application domain, and that focus on the **transformation, transport and processing** of content relevant to an application. Tables 3 and 4 classify papers contributing to these processes, and situate them into LMU-N. Section 3.1 details the processes of this category.

#### 2.3.2. Node-related processes

Node-related processes are activities not specific to an application domain, but common to the IoT domain. The development of such processes goes against vertical fracturation. The messages exchanged in these processes focus on the description of nodes characteristics, capabilities, preferences, etc, in order to allow nodes to be **aware** of their neighbours on the graph, and to offer a **homogeneous** representation. Table 5 references the papers contributing to these processes, and they are described in section 3.2.

Table 1  
LMU nodes physical characteristics

	Energy supply	Processing power	Communication capabilities	Memory	Storage
<b>Upper</b>	Traditional power grids	High to very high	Standard web protocols	High to very high (Several Go)	Large to very large (internal HD to disk bay)
<b>Middle</b>	Mixed, dependant on the deployment	Medium	Extended, both ad-hoc and standard web protocols [4]	Medium to low (Up to 1Go)	Medium to limited (Internal HD to SD card)
<b>Lower</b>	Often limited: battery, renewable source [19]	Very limited, often microcontroller	Constrained, Ad-hoc, potentially short range (BLE, Z-Wave) [4] [19]	Very low (Under 500Ko)	Limited (Flash memory, SD card)

Table 2  
LMU nodes roles and examples

	Role in the network	Example
<b>Upper</b>	Content processing, decision making, and user interface (display or API)	Cloud server, laptop, tablet
<b>Middle</b>	Local content aggregation, adaptation of ad-hoc communication to the web	Domestic box, gateway, micro-pc
<b>Lower</b>	Link to the physical world : measurement and action	Sensor, actuator

### 2.3.3. Ontologies of the SWoT

In order to integrate IoT deployments in the SWoT, their resources must be described using ontologies. Depending on the type of process, the ontologies required for the SWoT integration is different.

In the case of content-related processes, there is a need for ontologies capturing domain knowledge. The LOV4IoT<sup>11</sup> initiative mimics the principle of the LOV<sup>12</sup>, and references ontologies that have a connection with the IoT domain, either describing IoT concepts, or domain knowledge in a field impacted by the IoT, e.g. agriculture, home automation, or e-health. These ontologies are diverse, and disconnected from each other: the vertical fracturing between domains is not bridged.

Node-related processes are more focused on interoperability, and require the modelling of IoT systems common characteristics despite their diverse domains of interest. IoT ontologies capturing these connected device network characteristics exist. Several ontologies have been proposed in order to describe nodes:

Semantic Sensor Network (SSN)<sup>13</sup>, saref<sup>14</sup>[35], iot-ontology<sup>15</sup>, IoT-lite<sup>16</sup>[36], Spitfire<sup>17</sup>, IoT-S<sup>18</sup>, SA<sup>19</sup>, the oneM2M base ontology<sup>20</sup> and IoT-O<sup>21</sup>. Papers like [22], prior to the apparition of SSN, or [1], much more recent, survey these ontologies.

In both cases, the diversity of ontologies at stake can be an issue. That is why good practices must also be followed in the conception and the integration of these ontologies, described for instance in [5] or [18].

To sum up, LMU-N is an architectural pattern built from a bottom-up study of IoT architectures. It is composed of nodes and messages flows, two abstractions that can be described using ontologies. LMU-N is not the core contribution of this paper, but it provides a reading grid that is used in the next section. It is a relevant classification framework: 72% of the contributions listed in section 3 can be at least partially situated within LMU-N.

<sup>13</sup><http://purl.oclc.org/NET/ssnx/ssn>

<sup>14</sup><https://w3id.org/saref>

<sup>15</sup><http://ai-group.ds.unipi.gr/kotis/ontologies/IoT-ontology>

<sup>16</sup><http://iot.ee.surrey.ac.uk/fiware/ontologies/iot-lite>

<sup>17</sup><http://sensormeasurement.appspot.com/ont/sensor/spitfire.owl>

<sup>18</sup><http://personal.ee.surrey.ac.uk/Personal/P.Barnaghi/ontology/OWL-IoT-S.owl>

<sup>19</sup>[http://sensormeasurement.appspot.com/ont/sensor/hachem\\_onto.owl](http://sensormeasurement.appspot.com/ont/sensor/hachem_onto.owl)

<sup>20</sup>[http://www.onem2m.org/ontology/Base\\_Ontology/](http://www.onem2m.org/ontology/Base_Ontology/)

<sup>21</sup><http://www.irit.fr/recherches/MELODI/ontologies/IoT-O>

<sup>11</sup><http://www.sensormeasurement.appspot.com/?p=ontologies>

<sup>12</sup><http://lov.okfn.org/dataset/lov/>

## 2.4. Related work

Previous work has been done to survey the convergence between the IoT and the semantic web:

- Early work in the SWoT focused on semantic sensor networks. For instance, [22] surveys sensor ontologies and observation representations. The scope of this work is especially on models, and even if it proposes an overview of technologies enabling semantic sensor networks, it does not present specific applications. Similarly, [1] gives an overview of the semantic web stack applied to the IoT, and surveys IoT ontologies. It goes beyond semantic sensor networks, but is still limited to models analysis. We propose in this paper to focus on how the ontologies and the technologies of the semantic web are used to develop the SWoT, rather than on identifying exhaustively the models used.
- [37] is a survey of the IoT domain, proposing a definition for the notion of IoT and listing application domains and enabling technologies for the IoT. The IoT paradigm is described as the convergence of Internet technologies, electronic devices, and semantic web technologies. However, the paper itself does not cover how semantic web technologies are integrated into the WoT, while we intend to analyze in detail and compare different contributions to the SWoT.
- [38] studies the roles semantic web technologies can play in the IoT, as well as the challenges they represent. This paper identifies some processes similar to what is presented in this paper in section 3. However, we situated the contributions to these processes within LMU-N in order to enable a deeper understanding of how an IoT network is integrated into the SWoT, and to give a finer-grained analysis grid: we want to identify the impact of the nodes on the processes they are involved into. The explicit identification of processes, as well as the characterization of nodes based on identified criteria, also **allows future studies to be classified** within the analysis grid we propose.
- [39] gives an overview of the evolution from the IoT to the WoT and toward the SWoT. It is focused on the role of standards in interoperability, and the integration of semantic web technologies in standards. It also provides an overview of technologies at stake in the IoT. This paper is oriented

toward projects and industrial consortiums, which is out of the scope of our study. We focus on the contributions of the SWoT to IoT issues, and only integrate standardization concerns when they are related to this domain.

## 3. Contributions of the semantic web to IoT processes

In this section, we describe for each process the contributions of associated papers to point out the added value of semantic web principles and technologies. Each category of process (content-related and node-related) are separated into subcategories of general recurring patterns among processes in order to cluster them into coherent sets: complementary processes, or processes implicated in similar scenarios.

The complete classification is summarized in tables 3, 4 and 5. In these tables, each column represents a process, and each line represents the situation of the contribution in LMU-N: if applicable, the contribution is precisely situated between a source and a target node type. However, due to the lack of information in some paper, the positioning of some contributions is only partial (in the lines "unspecified"). Moreover, some contributions are described completely decorrelated from any architecture, or in an architecture not compatible with LMU-N, they are placed in the "Non-LMU/Not specified" line.

### 3.1. Content-related processes

Content-related processes classification is summarized in tables 3 and 4. Three main categories of content-related processes are distinguished, shown in the headers of the tables: representation transformation, transport, and processing, each described in the remainder of this section.

#### 3.1.1. Content representation transformation

These processes are dedicated to the transformation of content according to the DIKW hierarchy. **The core meaning of the content is not changed** by these processes, but the expressivity of its representation varies.

*Enrichment* is the process of transforming content representation upward in the DIKW hierarchy. For instance, the description of data with meta-data to transform it into an information is an enrichment: content is described with ontological entities to unambiguously define its meaning, capture its context and increase its

reusability and its overall value to an application. Enrichment is one of the earliest processes that used semantic web technologies to contribute to the IoT with contributions such as [16] or [29]. This matter of fact is explained by the predominance in early IoT work of sensor networks, a subset of IoT networks where all the LN are sensors. Enrichment is performed when content is exchanged upward in LMU-N: all the enrichment contributions in table 3 are directed upward, which makes enrichment an upstream process, increasing content complexity and processing cost.

In [29], the enrichment is performed by **annotating the content** with RDFa<sup>22</sup>. This approach does create semantically enriched content, but it requires the LN to produce complex, structured content (here compliant with the OGC's O&M model<sup>23</sup>). This might be impossible for some LN, as it requires to exchange heavy XML documents over communication links that can be very restricted (LPWAN networks or CoAP for instance). To overcome this issue, [23] proposes to **push the enrichment task to the MN level**: the LN produces raw data, in any format, this data is transported to the MN by a dedicated network, and only then the MN uses its own knowledge of the producing node to enrich the data. [40] instantiates enrichment at the same level as [23] (as shown on table 3), from LN to MN. However, contrary to [23], [40] proposes an approach where the LN offers a "semantic interface", implementing enrichment functionalities: content is enriched on the LN side. Unfortunately, the paper does not give any additional precision to describe its enrichment approach. In a different approach, enrichment to a semantic format is performed in [6] and [18] on the UN side when data is collected from the MN. Content collected from sensors is stored in a standardized structure on the gateway, but it is explicitly described with an ontology only on the server side.

The importance of the enrichment process is underlined by the important number of publications contributing to it: 17 papers in this survey. It comes from the discrepancy between the LN, hardly able to produce high-level content, and the UN having applicative need for homogeneous content represented in expressive formats. Moreover, transforming data into knowledge is a common practice in the semantic web domain, even outside the IoT. This is an explanation to the fact that 47% of the contributions to this process

cannot be situated precisely into LMU-N: their approaches are more centered on content than on the architecture around it.

**Lowering** is the process exactly **opposed to enrichment**: it is the transformation of content from a semantically rich format to a less expressive, more constrained representation. The need for such transformation arose from the integration not only of sensors, but also of actuators as LN in the IoT. These devices are content consumers, but their constrained nature prevents them from being able to consume some type of content. Actions represented in rich formats by UN need to be adapted to the target device so that it can interpret it and act as intended. This transformation deprives the content from part of its expressivity and context, but the transformed, simpler content is interpreted in a known context, leading to a **trade-off for consistency**. Lowering requires a node to have a representation of the capabilities and expectations of a remote node: it is a process that needs the source node to be more powerful than the destination node, and that is necessary because of the restrictions of the destination node. It is therefore a downstream process.

Lowering is opposed in the literature to "lifting", a synonym for enrichment. [41] describes SAWDSL, a language aiming at making it possible to lift XML to RDF and to lower RDF to XML thanks to XML schema annotations. However, in the studies surveyed here, most of the approaches to lowering are not explicitly based on annotations, but rather on ad-hoc approaches. For instance, in [18], an autonomic control loop is instantiated, and high-level representation of actions are transformed into service calls, but the transformation process ad-hoc: RDF individuals and their OWL descriptions are consumed by a software that call the procedures based on an a priori interpretation of the expressed content.

Overall, the transformation of content from a high-level representation to a form that can be processed by a constrained node is still a challenge for the SWoT. In the publications referenced in this paper, the proposed mechanisms are either **manual or ad-hoc**. A majority of the studies cannot be clearly situated in LMU-N, denoting shallow contributions. The lack of interest in this process comes from its contradiction with the usual practices in the semantic web community. Content is generally moved upward in the DIKW pyramid, because it gains value this way. However, the presence of **constrained content consumers** (actuator nodes) on the IoT, combined to the need for high-level con-

<sup>22</sup><https://www.w3.org/TR/xhtml1-rdfa-primer/>

<sup>23</sup><http://www.opengeospatial.org/standards/om>



tent in UN, makes lowering a process as necessary as enrichment for a real SWoT.

### 3.1.2. Transport

In processes listed under the transport category, content itself is not transformed. These contributions focus on how content is distributed across an IoT network (e.g. routing), or on how the content is used (e.g. control). Content itself is not necessarily expressed using the semantic web technologies, but its management on the network is based on these technologies.

*Notification/Dissemination* is the process of forwarding a piece of content to a distant node **in a push manner** (initiated by the content collector). It can be driven by the interest of the receiver, or on applicative logic of the emitter. Content is disseminated from the node that generated it, either by collection (typically by a LN), or by a higher-level process (e.g. enrichment by a MN or processing performed by a UN) toward another node. The diversity of nodes being sources or targets for this process makes it, in theory, a mixed process, going both upstream from LN and downstream from UN. This mixed aspect is present for instance in [31]: on the one hand, sensor data is sent upstream, toward a control center. The control center processes the data, and produces alert messages that are disseminated downstream.

[42] proposes a dissemination approach based on an ad-hoc interest expression for content consumers represented by a query. Each consumer query is registered on a content matching component in the form of predefined triple pattern. Triples matching a registered pattern will be redirected to the relevant content consumer. However, this approach is oriented toward performance, and reduces the expressivity of queries, limited to a unique triple where each element can be fixed to a constant value. Approaches based on stream processing, such as [33] or [43], use the full expressivity of SPARQL coupled to specific streaming operators, based on a time window or a number of events. The result of a stream query is an event stream, which is directed toward the querying node.

Even if fundamentally, dissemination is a mixed process, in the surveyed papers notifications are mainly sent upstream. This can be related to the architectural tendency of IoT networks to a hierarchical organization, with complex processings performed in the upper nodes. Therefore, it makes sense that UN, driven by applicative needs, express their needs to nodes, lower than them, that will collect content in a general purpose approach. The absence of goal-awareness in con-

tent collection by LN is motivated by the will to **avoid stovepiped deployments**.

*Control* is the process where a node sends a command to a remote node for it to execute. Commands are either issued by the user, or by high-level applications executed on UN, and are eventually executed by a LN, making control a downstream process. In this process, the use of semantic web technologies and principles allows the target node to have a deeper understanding of the command.

[44] propose an approach based on rule evaluation: if certain conditions (expressed in RDF) are matched, the rule entails an action. This approach allows the same node to have different behaviours upon the reception on the command, if the rule head relies on agents contextual knowledge. In [18] and [6], a representation of the actions to be taken is inferred from the representation of the user requirements and from the observations of the environment. These actions are associated to service description that are processed by a remote node to actually make the service calls.

The necessity for the control process arises from the integration into IoT networks of **actionable nodes**. The reduced number of actionable nodes deployed compared to sensors explains that only three papers discuss contributions to this process.

*Routing* is the process where a route among nodes is determined for a content instance. Routing policies are at the core of the internet, and not only of the IoT. The integration of the semantic web technologies enables the definition of new routing policies. Routing policies are not specifically directed toward a direction in LMU-N, therefore routing is a mixed process. Routing is different than dissemination in the sense that it is implemented by nodes on content they did not necessarily generate. Moreover, routing tends to be driven by network topology and nodes description, whereas dissemination by content characteristics.

[45] proposes a distributed approach to routing, where each node computes the optimal route from itself to a destination broker. Each node knows the distance of its neighbours to the destination broker, and to break ties, the semantic distance between the node profile and the profile of its neighbours is used. A node profile instantiates a dedicated ontology, and it is serialized into a binary string. Each node profile is unique, and used as an identifier. The evaluation function computes the energetic cost of the produced routing tree, and shows a clear **reduction of energy consumption**. In this approach, the semantic web principles are used

to reduce energy consumption at runtime. However, the algorithm used to serialize the semantic description of the node output is not clearly described. It outputs results that are suitable for constrained nodes, but it is unclear whether it supports the full expressivity of the ontology or not. In a different, hierarchical approach, [46] describes the construction of routing tables for service provisioning based on node clustering. Servicing nodes are described with an ontology, and they are clustered by the gateway they are connected to based on a semantic similarity measure. The obtained clusters are provided to the upper-level gateway, which uses the clusters from the nodes to build a routing table, and reproduces the process recursively. [47] also uses a semantic description of sensor nodes to create clusters of nodes satisfying a certain query that represents the interest of the user, for instance "Nodes detecting a temperature over 50 degrees". These clusters then aggregate the content they collect, before sending it to a sink node. The authors used the number of nodes alive against time as a metrics to show the effectiveness of their approach: the economy of energy induced by the reduction of communications allows nodes to communicate longer. However, the authors do not detail how semantic technologies are used precisely, and they make the assumption that the nodes are sufficiently powerful to match their own profile against a user query.

In these three approaches, the routes to forward content from a node to another are computed based on a semantic description of nodes. However, to the best of our knowledge, no contribution was made toward the integration of knowledge about the content itself in the routing process.

*Querying* is the process where an application explicitly accesses content on a remote node, in a **request/response manner**.

On the IoT, content might not necessarily be enriched and stored in a knowledge base, preventing it from being queried using semantic web technologies. [43] proposes a streaming Ontology-Based Data Access (OBDA) layer to access data in a mixed approach, where stream semantic queries are processed through R2RML mappings to be transformed into queries over federated sensor networks, and to transform the response in order to answer the original SPARQL stream query. [48] proposes a similar approach, with transformation from SPARQL to SQL via RML, with and without the streaming extension to SPARQL. In other papers, where querying is not the core contribution,

direct querying in SPARQL is also used after storing RDF content in a knowledge base, as it is proposed in [12] for instance. [31] proposes a mixed approach, where part of the data is stored in a traditional SQL database, but its associated metadata is stored as RDF for more expressive SPARQL queries.

The request/response interrogation of RDF graphs using SPARQL queries is **not specific to the IoT**, and that is why few contributions in the survey cover it. This is also a reason why none of the contributions for this process can be described precisely with LMU-N: the focus is not on the IoT architecture. Transformation from SPARQL to SQL is not the main practice to query IoT content, contrary to what this survey may suggest at a first glance: it is a research topic receiving contributions, as it is presented in section 4.2.2, dedicated to the evolutions of the semantic web.

### 3.1.3. Processing content

Processing is an important part of the IoT value creation, as is described in [49]. It covers a broad range of processes (we identified 5) where content is leveraged in an application, and semantic web principles and technologies can be used in order to provide innovative services.

*Abstraction* has similarities with Complex Event Processing (CEP), as [50] points out: low level symptoms are extracted from content and correlated together in order to be transformed into a more abstract diagnosis. It can be based on reasoning, rules, pattern-recognition, etc. Abstraction requires both the collection of content, and resource-intensive processing, it is therefore an upstream process. Abstraction is a process already present in early studies such as [29] or [30], where an abstraction is defined as the "representation of an environment derived from sensor observation". It is a process that has been widely implemented: 11 papers in this survey are dedicated to abstraction. Each contribution will not be presented in details in this paper, but they all have common characteristics: they use content described with an ontology (ssn for [51], or SenML for [52]), and infer new content described with a domain-specific vocabulary. The production of abstract content is necessarily **driven by an applicative need**.

[29] proposes an SWRL-based approach to infer new content from sensor observations, illustrated by the example of the deduction of road conditions from temperature and precipitations observations. In [30], the authors propose a logic-based approach to abstraction, and uses OWL-DL reasoning to derive the most

probable abstraction from a set of observations. [11] takes another approach to abstraction, proposing a survey on context-awareness in the IoT, and how semantic web technologies can support the modelling of context, as well as contextual reasoning. The authors also propose several steps to abstraction, with a granularity that could vary according to the layers involved in the process: content transferred from LN to MN is processed a first time to get a primary context, and it is processed a second time for more complex treatments on larger content instances when transferred to UN.

Inferring new knowledge from a KB, optionally combined to a set of rules, is a logical process not limited to the SWoT. Traditionally, it is performed horizontally on a powerful machine: the KB, the reasoner and the means to insert new content in the KB are all hosted on the same server. However, in the case of the IoT, the process is not horizontal on UN, but rather upstream, and performed on the UN side. This is related to the nature of inference in IoT applications: it is **fed by content produced by LN**.

*Aggregation* is a process where multiple instances of content are used in order to produce a new content instance of the same level in the DIKW hierarchy, as opposed to abstraction where the inferred content is of a nature different from the pieces of content used for inference. For instance, computing the average (or the maximum) of several values is aggregation, while using the same values to infer the occurrence of a meteorological event is abstraction.

In [53], the aggregated elements are similarity matrices used for ontology alignment. The authors tackle the heterogeneity of application fields for the IoT domain, and the diversity of ontologies it entails, by proposing an ontology alignment technique. The proposition of these authors is not designed to be run online, and therefore it does not fit the LMU-N reading grid. The proposed aggregation approach, driven by a consistency measure, is applicable to any content annotated with an ontology. In [20] and [33], aggregation of content is performed using operators from SPARQL, like COUNT, MIN, etc, from a customized extension of SPARQL adapted to streaming queries to perform aggregation over time or space. In these papers, aggregation is a driving mechanism for sensor mash-up. However, content aggregation is not separated from content enrichment for instance, these two processes being both covered by the term "data fusion" in [20]. In these papers, the aggregated content is both pro-

duced and used in UN, making aggregation a horizontal process.

In the papers of the survey, data fusion is not represented as an upstream process, which is counter-intuitive. It is explained by the fact that all the papers relevant to the survey perform fusion operations on content **after it has been enriched**. Therefore, the content is enriched in an upstream process, but no contribution we surveyed performed the enrichment from lower to middle nodes, and then enrichment from middle to upper nodes. In this hypothetical case, fusion would be performed upstream.

*Visualization* is the display of content in a human-readable manner. When produced, IoT content is typically a numerical value with some metadata, and visualization processes propose a visual interpretation of it instead of its raw numerical form.

For instance, studies such as [29] or [54] use geographical metadata to display the location of content sources on a map, and to give access to these sources via a map interface. In [16], an additional processing is executed to represent not only the geographical location of entities, but also to extract further meaning from the content and to attribute a color code to the sensors depending on their nature (i.e. the class they instantiate in an ontology). In [33] (an extension of [54]), content is aggregated prior to the visualization phase, in order to display refined content, like heat maps or graphs, complementary to the map information.

*Consistency enforcement* is the process of ensuring the consistency of the content, that is to say the absence of contradictions among the facts expressed in the content, and the absence of wrong assertions (e.g. sensor measures different from the reality of the physical world).

[11] discusses the role of context in the identification of inconsistency issues. [55] proposes an ontology design pattern to capture the semantics of an industrial, non-standard, tagging vocabulary defined for the project Haystack<sup>24</sup> with the semantic web formalisms. The obtained ontology is then automatically populated using the instances of Haystack tags to generate the corresponding individuals. However, the transformation from a vocabulary with limited expressivity to a rich ontology can lead to inconsistencies, for instance if individuals bear two tags from the vocabulary that are disjoint in the ontology. The authors use DL rea-

<sup>24</sup><http://project-haystack.org/>

soning to detect inconsistencies in the generated KB, and identify modelling issues in the ontology or inconsistencies in the individual tags.

Consistency is not instantiated in details in the contributions we surveyed, even if [9] identifies consistency as a challenge for the SWoT. Contradictions between several nodes can be dealt with in an aggregation process where data fusion techniques are applied, but consistency enforcement also includes the detection of logical issues in models and their instantiations. More generally, consistency enforcement is not a topic limited to the SWoT, and **generic consistency mechanisms could be applied to IoT datasets**.

*Decision support* is a process where content is used as the input of a decision-making process. Some smart city scenarios use sensor networks and IoT deployments for decision support, e.g. Dublin's Smart Energy Demand Analysis<sup>25</sup>.

[31] proposes an Early Warning System (EWS) architecture enabled with semantic functionalities. Content collected by sensors is enriched and disseminated upstream, before being used by UN in order to be abstracted into high-level events. Once the events identified, the system takes operational decisions (where to send rescue, what places to evacuate...), and these operational decisions are implemented in downstream processes. [56] proposes the modelling of policies by ontologies in order to manage the services offered by diverse nodes, dynamically created according to context. The policies capture directives for decision making.

Decision support is a process with two aspects: on the one hand, the supported decision-making process can be performed by human beings, and on the other hand it applies to autonomic systems. The former case is not specific to the IoT, but can be instantiated with a sensor network, and the system only provides guidance to a human decision maker.

### 3.2. Node-related processes

In node-related processes, the messages exchanged between nodes do not focus on the content these nodes collect or process, but on the nodes themselves. Node-related processes classification is summarized in tables 5. They are separated in two sub-categories: processes dedicated to awareness among nodes, and processes dedicated to node heterogeneity management.

#### 3.2.1. Process related to neighbourhood awareness

In order to communicate, IoT nodes need to **be aware of the existence of each other**, and to have respective addresses to exchange messages. Furthermore, nodes may be only intermittently available, to save battery life for example, or due to failure, maintenance operations, etc. In this context, a dynamic awareness of a node's surroundings is required, and the following processes contribute to it.

*Discovery* is the process of gathering descriptions of the nodes that can be reached from the node host of the process. In a typical hierarchical IoT architecture, nodes of a given level are connected to multiple nodes of an inferior level, and to a few nodes of superior level (often only one). Therefore, LN can be deployed with the address of the middle node they communicate with hard-coded, while MN or UN need to discover underlying nodes as they come and go. That is why discovery is a downstream process.

[34] proposes a discovery method based on google's Physical Web (PW)<sup>26</sup> and extending it to a so-called Physical Semantic Web (PSW). Discovery of nearby nodes is enabled by the protocols and technologies of the PW, but the PSW extracts from the identification messages semantic annotations in order to decide whether the node is relevant or not to the client, running on a mobile device. [40] proposes a discovery for MN based on a semantic description issued to a central repository which address is known a priori by the issuer of the discovery request. Similarly, SPARQL query can be issued to the MN describing LN characteristics in order to collect LN identifiers. The ontology proposed in [57] is developed in order to ease node discovery by modelling contextual knowledge, such as location, domain knowledge, and policy knowledge. [58], [38], [32], [59] and [60] all propose a discovery based on a node specification sent to a repository containing nodes descriptions.

Discovery is a widely implemented process (10 contributions in this survey), because it is related to the dynamic nature of an IoT network: it allows **nodes' representation of the graph topography to remain consistent with the reality**. Discovery is also a process that is necessary to perform other processes. Discovering the capabilities of a distant nodes is required for processes such as lowering for instance.

<sup>25</sup><http://smartdublin.ie/smartstories/spatial-energy-demand-analysis/>

<sup>26</sup><http://google.github.io/physical-web/>

*Exposition* is the process complementary to discovery, in which a node makes its own description available in order to be discoverable. However, **exposition is initiated by the node target of the discovery**, that is why it is an upstream process.

In [34], exposition is supported by the PSW, already described for discovery. In [10], the authors propose to enhance the registration mechanisms offered by the oneM2M standard to support a semantic description exposition. A LN can register itself onto a MN if it knows its address, and the registration query contains a standard description of the LN. The authors propose to include a semantic description, or a dereferenceable URI, in the registration request. Since then, the oneM2M standard included a "semantic descriptor" resource, carrying an RDF/XML description of its parent node. It allows a node to expose its capabilities to its target when registering onto it. Another aspect of exposition is **proxying**: after having discovered a set of LN, a MN can act as a proxy and perform exposition of said LN in their stead, as [24] proposes. The authors propose an architecture where the exposition/discovery process can be specific to a type of network or technology. This way, the process is both adapted to the constraints of the LN and understood by the MN, which performs an enrichment phase on the node metadata in order to redistribute them in a more generic format.

All the contributions implementing these processes can be precisely situated in LMU-N, because they all specifically describe technical content with a detailed architecture. There is significantly fewer contributions to the exposition process than to the discovery process, which seems to be counter-intuitive regarding the dependency of discovery upon exposition. However, exposition is a process requiring the lowest node to be active, while in discovery, the burden is on the highest node, making it easier to implement.

*Selection* is a process where a node decides which other node should perform a task. It is especially relevant when multiples nodes are offering similar services yet having different capabilities, characteristics, costs, etc. Node selection is a downstream process, driven by high-level policies (e.g. energy saving or time efficiency), based on criteria to compare nodes, enabling the computation of a score and ranking. The use of semantic web technologies and principle are used to define these criteria.

[25] refers to the node selection process as the assessment of relevance metrics. The authors use ab-

stracted nodes, or Virtual Object (VO), and the semantic description of their characteristics (not precisely described in the paper), in order to perform a selection. The proximity criteria are defined dynamically depending on applicative requirements. This work focuses on the ability to perform a selection despite the heterogeneity of underlying objects thanks to the abstract representation of nodes. In [61], the authors identify two types of criteria for node selection: non-negotiable criteria, representing the ability of the node to provide a service, and negotiable requirements, over which selected nodes will be ranked. The filtering phase is performed using SPARQL queries representing user requirements, and the ranking phase is based on a multi-dimensional criteria aggregation. [62] describes a service selection process to create a composite service out of existing services. The actual node selection is performed at binding time, where services descriptions stored in a service cache by the middle nodes is used to associate the composite service with actual nodes. In [46], node selection is a process driven by a service search initiated by the user. This search starts from a top gateway which is the entry point of the user to a building network. Node selection is performed recursively in a gateway hierarchy, based on the semantic description of services clusters. At the lowest level of the hierarchy, the final gateway, actually connected to the lower nodes, returns the actual characteristics of the nodes matching the service request. In [44], the node selection is based on an a priori prediction of the entities that would potentially match a user query: only sensors that are associated to these entities are queried.

Similarly to discovery, selection is a downstream process where the highest node perform most (if not all) the computation. Node selection can be performed to initiate a notification process: in the case of upstream notification, the lowest node has to decide whether to send a content to an upper node or not. To reduce the load on the lowest node, the process can be switched into a node selection, where the upper node selects appropriate lower nodes to provide it with content. However, in this approach, one can see that the interest is switched from the content description in notification to the node description in node selection.

### 3.2.2. Processes providing homogeneity among nodes

As stated in the introduction, heterogeneity, and the interoperability issues it brings, is one of the leading reasons for the introduction of semantic web technologies into the IoT. The three processes dedicated to ho-

mogeneity are focus on the management of this diversity.

*Abstraction* is the process of representing a node by a virtual entity. In a context of heterogeneous nodes, abstraction aims at focusing on the modelling of **generic nodes representations to describe their characteristic in a unified way**. It allows applications to deal with a set of homogeneous nodes, breaking the vertical silos between application domains. Nodes manipulate abstracted representations for nodes of inferior level, making abstraction an upstream process. On the SWoT, **ontologies are used to describe abstract nodes**.

In [12], the behavioural patterns of nodes are used in order to automatically attach a description to unclassified nodes. Similarity is computed between the output of the nodes already described in the KB and those of the newly introduced nodes, and the undescribed nodes is annotated using the description of the similar nodes. This approach suffers a cold start issue: when only a few sensors are annotated, the clustering is less efficient, and the system needs to grow in order to propose a wider variety of sensors and more representative clusters. [25] proposes the notion of VO to describe how nodes can be abstracted, and describes how this approach tackles heterogeneity issues. The authors then describe how abstracted nodes can be used in other processes, such as composition or selection. These contributions are described in the corresponding paragraphs. In [15], physical nodes are associated to avatars, a virtual representations described in OWL. The authors identify requirements for a WoT platform, and show how their proposed avatar architecture meets these requirements, such as interoperability, reactivity, safety... Avatars are also given introspection capabilities, in order to support collaboration and service composition, described more specifically in the paragraphs associated to these processes.

Abstraction is an important process, contributed to by 14 papers in this survey. Among the processes situated in LMU-N, more contributions are situated from lower to middle nodes than from middle to upper nodes. A possible explanation is that the **LN class is the most heterogeneous**: end devices have the widest variety of functionalities. Therefore, being able to abstract their representation as soon as possible in the IoT hierarchical network makes their management easier for MN.

*Composition* is the process of associating nodes between them in order to create new nodes, offering services that were unavailable on the network before. It is a process often connected to node abstraction, because an abstracted node representation enables the creation of virtual composite nodes.

The notion of VO presented in [25] and in [13] is associated to the notion of Composite Virtual Object (CVO). The authors of these contribution describe a process to **build CVOs on top of homogeneous VOs**, themselves being abstractions for real-world objects. Based on a specification of applicative needs, CVOs are dynamically created. In [62], node composition is performed using the semantic description of different services. A plan is then computed, and services selected are called sequentially according to the plan. The process is dynamic, and the composite nodes described by the execution plan is not stored in the KB. [57] uses the same terminology of service offered by a CVO composed of several VO. They propose a modular ontology architecture, and associate the ontology modules with the entities they define: only the VO are described using the Resource ontology module, but it is the CVO and the services that are annotated with the Service ontology module for instance. However, the authors do not give insight regarding the creation of the CVO.

Composing nodes is only possible provided that these nodes are described by abstractions, and that these abstractions can be manipulated. This is why a majority of papers are present in both abstraction and composition process: they first define how they abstract physical nodes, and then demonstrate how these abstractions can be manipulated separately from the nodes they initially represent. The semantic web principle and technologies provide flexible models that support such approach.

*Specification/Configuration* is the process opposite to abstraction: generic representations are adapted to specific deployments, and physical nodes are configured in order to match their virtual representation.[44] proposes a different definition of configuration, focused on node introspection and pattern learning: new nodes cluster with existing nodes in order to automatically annotate the data they produce. It does not qualify as configuration as it is defined in the present paper.

[40] proposes an architecture where smart agents consume semantic messages from a broker, and control actuators by changing their virtual representation. This process is different from control as defined

in the content-centric process because the focus is not on the message that is sent in order to change the state of the device, but rather on the representation by the system of the state of the device as it should be. However, the paper does not provide further details on the implementation.

Globally, specification process as the lowering of a generic, virtual representation into potentially several implementation-specific commands has not been implemented, to the best of our knowledge.

### 3.3. Identified trends

Four architectural characteristics emerge from existing studies:

- (i) Communication between nodes of different levels are limited to adjacent layers. Processes being supported by message flows, processes instantiations are therefore specifically represented in the tables 3, 4 and 5 between nodes of adjacent classes, e.g. from upper to middle node.
- (ii) A node of a given level has contacts with a limited number of higher level nodes, and multiple contacts with lower level nodes.
- (iii) Horizontal contacts between nodes of the same level multiply as the level of said nodes gets higher: there is little to no direct contact between lower nodes, and upper nodes communicate with each other frequently. The five contribution to horizontal processes presented in the tables are all instantiated between UN.
- (iv) The IoT being a system tightly coupled with the physical world, nodes differ by their "proximity" with the environment. For a given node, this characteristic can be measured by the number of edges that separate it from a node having a direct contact (sensing or acting) in the environment. As one would expect, the lower a node is in the LMU-N hierarchy, the closer it is to the environment: LN directly observe or act on the environment, while UN only manage a digital representation of the physical world.

Situating SWoT contributions in LMU-N shows how **nodes constraint the processes they interact with**: contributions dedicated to the same process but at different levels in LMU-N do not expect similar outcomes. However, it is worth noting that some contributions are only partially situated in LMU-N because the papers describing these contributions do not provide enough implementation details. This either reveals that

Table 3: Papers contributing to LMU content-centric processes (1/2)

	From	To	Representation transformation		Transport/ Provisioning		
			Enrichment	Lowering	Notification/ Dissemination	Control	Routing
Upstream	L	M	[23], [4], [40], [63]				
	M	U	[20], [6], [18]				
Downstream	Unspecified		[29], [32], [64], [54], [33], [12], [22], [65]		[42], [54], [33]		
	U	M		[18]		[6], [18]	
	M	L		[40]			[46]
	Unspecified			[44], [66], [65]		[44]	[49], [54]
Horizontal	U						
	M						
	L						
Non-LMU/Non specified	Mixed				[31]		
			[43], [55], [67]	[55]		[45]	[43], [47], [48]

		Awareness			Homogeneity		Specification /Configuration
From	To	Discovery	Exposition	Selection	Abstraction	Composition	
Upstream	L		[10], [34]		[14], [24], [40], [46]	[14]	
	M		[70], [24]		[13], [6]	[20], [13]	
	U				[58], [61], [25], [15], [57], [66], [12]	[25], [15], [57], [62]	
	Unspecified						
Downstream	U						
	M						
	L						[40]
	Unspecified						
Horizontal	U						
	M						
	L						
Mixed							
Non-L/M/U/Unspecified							

		Processing				
		Abstraction	Consistency enforcement	Aggregation	Visualization	Decision support
Upstream	From L	[11] [11], [6], [52], [63], [51] [50], [38], [49]				
	To M					
	U					
Downstream	U					
	M					
	L					
Horizontal	Unspecified					
	U			[31], [20], [33]		[31]
	M					
Mixed	L					
Non-L/M/U/Non specified		[29], [30], [67]	[11], [55]	[53], [68]	[29], [16]	[69], [56]



the deployment into a concrete architecture is not in the scope of the paper, or that the process presented is not specific to the IoT but rather uses the IoT as an application domain, for instance in the case of the contributions to the querying process. The constraints brought by the nodes justifies the presence of [40] twice for the discovery process: discovery from upper to middle nodes being differs from discovery between middle and lower nodes.

The superior capabilities of nodes of higher level allows system designers to **push the constraints brought by the limitations of lower nodes up**. This is why, regardless of the direction of the process, the highest nodes between the target and the source performs the more computation for content transformation and processing processes, as well as for node-centered processes. In both cases of discovery and enrichment, which are opposed processes, this observation is verified: UN are responsible for the discovery of MN for instance, and content produced by LN is enriched by MN. Moreover, architectural characteristic (ii) is a consequence of this shift of constraints: a middle node is able to manage the processing required for content enrichment that multiple lower nodes could not perform.

The influence of nodes on processes is not limited to physical constraints: the **highest nodes also holds the largest part of the application logic**. Having LN collecting content without a specific applicative goal limits its vertical fracturation: only the nodes processing content should be application-oriented. According to the architectural characteristic (iv), reducing vertical fracturation requires (among other measures) to reduce the application logic executed on nodes with a high priority with the environment.

A side effect of this is the predominance of upstream processes compared to downstream processes. When comparing enrichment to lowering, or abstraction to specialization, it is clear that **upstream processes are studied more actively**. An exception to this predominance is the case of discovery and exposition, clearly visible in table 5: the downstream process is more represented. This is coherent with the location in the highest nodes of the application logic and the heaviest processing. Our analysis of this matter of fact is the **similarity between upstream processes in the specific case of the IoT, and traditional research in the semantic web domain**: enriching content with metadata, or inferring abstraction from knowledge base are well-established topics. On the other hand, the need for downstream processes such as lowering of special-

ization arises from the constrained nature of some IoT nodes, and therefore the application domain is much more specific than the one of upstream processes. The predominance of upstream processes is coherent with the predominance of sensor nodes among IoT nodes: **downstream processes are required for actionable nodes**, which are a minority. The prevalence of upstream processes also relates to the need for homogeneity: when performing upstream operations such as node or content abstraction, the result is expressed in a rich model, providing an interoperable representation.

Moreover, in some surveyed architectures, centralization is an issue. Even if the IoT is by nature a distributed system, given its hierarchical topology, upstream processes concentrate high-level content in a few UN. When services (such as visualization) are offered directly by the server to its clients, this model is suitable, but when the network needs the capacity to make **autonomous local decisions**, this approach shows shortcomings. With the multiplication of nodes, and the increase data flow it involves, scalability is an issue for resource-intensive processes, and the next section surveys approaches to tackle this issue. Moreover, the lack of downstream processes limits the ability of MN to benefit from the content inferred by the UN that would enable them to make local decisions.

This section described contributions of the semantic web to the IoT, with the identification of generic processes and the situation of the research contributions toward these processes within LMU-N. Processes such as enrichment or abstraction exist in the semantic web outside of the IoT, and they are used to solve issues specific to the IoT. However, processes such as lowering or control are more specific to the convergence between the IoT and the semantic web, and are part of evolutions of the semantic web driven by IoT constraints. These evolutions are described in the next section.

#### 4. How the semantic web evolves to face IoT constraints

The issues tackled by emergence of the SWoT are issues affecting the development of the IoT and the deployment of processes within IoT architectures. These issues (e.g. heterogeneity, lack of interoperability, content transformation) are recurrent concerns for the semantic web community, not necessarily related to the IoT or the WoT. However, the IoT also has intrinsic

characteristics due to the constraints on its constituting nodes, the distributed nature of its deployments, and the dynamism of its topology. These constraints apply to any solution deployed in an IoT architecture. That is why **contributions between the semantic web and the IoT domain are not unidirectional**: the semantic web does contribute to the emergence of the SWoT, but semantic web technologies and principles must also be adapted to meet the constraints of the IoT in order to develop the SWoT.

#### 4.1. Adapting to the constraints on the nodes

One of the most important constraint in the IoT is the presence of constrained physical nodes, be it LN or, to some extent, MN. The constraints on these nodes were presented in section 2.2.1. Among the resources that can be limited for a nodes, we distinguished energy, processing power, communication channels, and memory. These resources are not independent, e.g. the limitation of processing power or of time exchanging messages over the communication channel saves energy. In order to be suitable for an IoT network, a solution should be adaptative regarding the node running it, and the integration of semantic web principles and technologies must be thought differently on each level of LMU-N: a LN does not have the capability to run the full semantic web stack.

Energy is a primary concern in the IoT domain, for two reasons. On the one hand, the dissemination of nodes in the environment makes it hard to connect them to a power grid (especially in non-urban area, such as fields or forests). Therefore, some nodes run on batteries, and their lifetime is directly related to their consumption of energy. On the other hand, the multiplication of nodes implies a multiplication of power consumers, and managing energy consumption at the scale of the node leads to energy saving at a more global scale. The use of semantic web technologies to reduce the cost of content transport was already discussed in the description of the associated processes presented by [47] and [45]. However, in these papers, the use of semantic web technologies is not modified to match the constraints of an IoT network.

To show how semantic web technologies can be adapted to IoT networks requirements, [72] compares the different serialization formats available for semantically rich data with respect to the size of the messages encoded in each format, as well as the number of CPU cycle required to produce these messages. The energy consumption associated to both the creation, the trans-

mission, the reception and the decoding of these messages is then compared for each format. In this paper, **two aspects of energetic consumption** are pointed out: the **processing** required to handle the content, and the **communication** channels required to exchange it.

Reducing the cost of the communication can be achieved by using protocols adapted to the needs and constraints of the IoT, and by adapting existing platforms to these protocols. For instance, [73] proposes an extension of the Linked Data Platform (LDP)<sup>27</sup> specification, a recommendation of the W3C. The W3C links primitives of LDP to HTTP<sup>28</sup>, a protocol rooted in TCP<sup>29</sup>, therefore requiring connection between the communicating entities. HTTP is therefore not adapted to IoT architectures, where more lightweight protocols are preferred. For instance, the authors of this paper propose a mapping of LDP to CoAP<sup>30</sup>, based on UDP<sup>31</sup>. CoAP is a protocol especially designed for constrained applications, with reduced headers and limited packet body. Such initiative allows IoT nodes to be connected to the LOD, and therefore to extend the WoT, while respecting the constraints of IoT nodes.

Distributing content via adapted protocol also requires said content to be stored on the device distributing it. Contributions such as [74] aim at allowing devices with limited memory to store semantically rich data. In this paper, the authors propose a method to store compressed RDF data in memory while ensuring a certain level of efficiency regarding encoding and decoding of data. This paper is not specifically targeted to the IoT domain, but its contribution matches the requirements of this domain. The ability to be able to access and modify efficiently the stored data is important in the case of streaming data, the authors point out. In the case of IoT deployments, and especially of sensor networks, the ability of storing and distributing up-to-date data is an important feature, and there must be a trade-off between memory optimization, and cost of encoding/decoding. [26] also proposes a tuple store suitable for embedded systems, as well as a protocol-independent RDF broker, that can be mapped to CoAP for instance. The authors propose the adoption of a protocol stack adapted to constrained nodes in order to

<sup>27</sup><https://www.w3.org/TR/ldp/>

<sup>28</sup><https://tools.ietf.org/html/rfc2616>

<sup>29</sup><https://tools.ietf.org/html/rfc793>

<sup>30</sup><http://coap.technology/>

<sup>31</sup><https://www.ietf.org/rfc/rfc768>

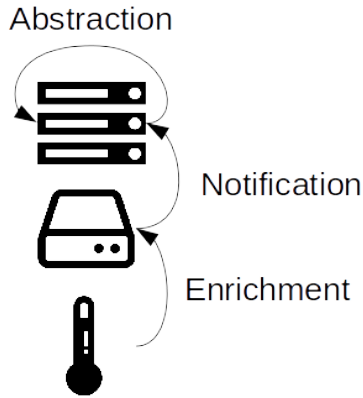


Fig. 3. Example of sequential processes

include them into the SWoT without the need for smart gateways acting as proxies.

Another adaptation to the constraints of the nodes is the **splitting of an application into processes**, and the spreading of these processes across the IoT depending of the nodes able to support it. This approach is at the core of the LMU-N architecture, and explains why some papers appear in the tables at different levels or for different contributions, such as [31] where the authors propose contributions to both aggregation and decision support. Figure 3 gives a very simple illustration of the sequential spreading of processes, similar to [4] for instance. The content generated by the LN is enriched by the MN, and abstracted according to applicative needs by the UN after it has been notified.

#### 4.2. Managing characteristics of IoT content

Besides the constraints on nodes, the IoT also produces content that has specific intrinsic characteristics. The usage of semantic web principles have to be adapted to this type of content as well.

##### 4.2.1. Stream processing

The content dealt with on the SWoT is diverse. The two families of processes identified in section 2.3 (node-related and content-related) are separated by the nature of the focus of the messages they convey, and the dynamism of the content these processes rely on varies. Node-related content has both a static and a dynamic part: endpoint or API description for instance do not vary often for a given node, but its battery level or its availability can be updated frequently. Content collected by IoT nodes, on the other hand, is associated to a time and a place of collection, and is by nature dy-

namic. **Applications relying on such up-to-date content do not need large historic storage.**

This behaviour is associated to **stream processing**, an approach used in papers such as [43], [75] or [33]. In these papers, part of the content, especially content describing the system itself, is static, while content generated by the system (e.g. sensor observations) is managed as an endless stream. This dual approach to content processing ensures scalability, due to the potentially infinite nature of the content stream generated by the nodes. SPARQL extensions, such as C-SPARQL<sup>32</sup> or CQELS<sup>33</sup>, are design to execute of continuous queries over RDF streams. Continuous querying is associated to windowing operators bounding the queried portion of the stream. [75] distinguishes two types of windows: time-based, containing all events over a period of time, and event-based, containing a fixed number of events. The repetitive, structured nature of IoT content makes it suitable for both windows types, the choice depending on the application.

##### 4.2.2. Ontology-based Data Access

OBDA is an approach to content access where the semantic web technologies are used in order to build an abstraction layer on top of a content source [68]. Queries are issued to the abstraction layer, and benefit from the expressivity of the semantic web technologies. The abstraction layer then transforms the query based on mappings to the destination content source language, retrieves content, and enriches content according to the mappings before returning them to the issuer of the query. This approach trades storage efficiency in large content instances for query language expressivity. Mapping languages such as R2RML<sup>34</sup> or more recently RML<sup>35</sup> propose a syntax to define mappings between a structured data format (relational databases for both, XML, JSON and CSV for RML).

[75] propose to access timestamped data streams using the SSN ontology to access streams of observations produced by sensors and stored in relational databases for the Swiss Experiment project<sup>36</sup>. This approach is motivated by the heterogeneity of the underlying data models for the different data sources. [68] proposes an OBDA approach where **static and streaming data are separated**, in order to identify clearly **background**

<sup>32</sup><http://streamreasoning.org/resources/c-sparql>

<sup>33</sup><https://github.com/cqels>

<sup>34</sup><https://www.w3.org/TR/r2rml/>

<sup>35</sup><http://rml.io/>

<sup>36</sup><http://www.swiss-experiment.ch>

**knowledge and varying observations.** Their motivation is also the homogeneity provided by the abstraction layer over the very heterogeneous underlying sensor network. The authors extend the role of the OBDA layer in their system, by adding content aggregation functions to the content enrichment. [48] details why IoT content is adapted to the OBDA approach: the observations produced by the sensors are strongly structured, and storing only one field header for all the observations is more efficient than storing the metadata for each observation. The mapping from the metadata to the schema can be easily stored in memory, and queries on traditional relational databases are more efficient than queries on RDF stores. The limitation of this approach, however, is a reduction of the flexibility of the schema compared to an ontology, and the possibility to reason over the dataset. It is worth noting that OBDA and stream processing can be complementary approaches, for instance in [43] and [75].

Overall, this section underlines how the convergence between the IoT and the semantic web toward the SWoT challenges the semantic web principles and technologies: the constraints on the nodes and specificities of IoT content are sources of innovation.

## 5. Conclusion, perspectives and future work

The IoT is a central scientific and technological topic, with a great expected societal impact. In an effort to solve interoperability issues, several IoT networks are being connected to the WoT, and semantic web principles and technologies are used to turn the WoT into the SWoT. In order to be able to compare the existing IoT architectures presented by the SWoT community, we proposed in this paper LMU-N, a unifying IoT architectural pattern issued from a bottom-up analysis. LMU-N describes a graph of nodes, where three types of nodes are separated based on their physical capabilities and their roles on the network. Generic processes have been identified through the projection of SWoT contributions on LMU-N, and the contributions of the semantic web to the IoT domain have been situated with respect to these processes and to LMU-N. Our unified approach allowed us to identify certain trends: upstream processes are predominantly represented over downstream processes, and the capacities of the nodes influence the distribution of processes across the network. To complement this study, evolutions of the semantic web, developed in particular to match constraints of the IoT, have been presented.

This twofold approach to the landscape of the SWoT showed the reciprocity of the convergence of the semantic web and the IoT: not only does the semantic web provide solutions to the interoperability and complexity issues of the IoT, but the IoT also challenges the semantic web principles and technologies to evolve to be compliant with its constraints.

Even if tackled by some studies surveyed in this paper, some issues remain open for scientific contributions. For instance, **consistency** of content across nodes networks has to be ensured, in spite of content transformations when it is enriched or lowered. While contributions to content enrichment are many, table 3 reveals a reduced number of papers in the domain of lowering, a process essential to consistency.

The IoT is a **constrained domain**, and it requires a continuation of the research presented in section 4. Formalisms adapted to constrained nodes and networks are emerging, proposed for instance by the W3C's WoT working group, and techniques to integrate legacy devices in semantic-aware networks need to be developed.

The increasing number of nodes is associated to an increase in the generated content volume, requiring studies such as those presented in 4.2.1 to be extended in order to tackle **scalability** issues. The use of decentralized approaches to LMU-N processes is also crucial to scalability: if sub-parts of an IoT network are able to collect, enrich, and process data in order to provide a service, the load of data exchanged on the global network is reduced, and decisions can be taken locally. Enabling local autonomic behaviours also increases resilience. However, local decisions can lead to the emergence of inconsistent global behaviours, as it is presented in the system of system theory[76]. Security, the non-functional requirement listed number one concern by open-source IoT developers in a recent survey by the Eclipse foundation[7], also needs to be integrated deeper in the SWoT in order to enable safe autonomic computing.

LMU-N will evolve to be adaptable to different architectural granularities, in an "unfolding" approach. In the case of very large scale deployments, such as smart cities, the node abstraction is not only suitable for devices and services, but also for self-sufficient networks. A smart building is a node compared to the city, and the capacities used for nodes classification needs to be adapted to be adapted to this recursive approach.

Furthermore, multiple perspectives are opened by the development of the SWoT. Among them, a promising opportunity is the evolution of IoT traditional machine-centric content to a richer, more expressive content, described with vocabularies connected to natural language resources.

These resources enable the next step in semantic interoperability on the SWoT, model alignment. Manual model mapping is possible in all cases, but automatic alignment requires ontologies' expressivity and machine-understandability. For instance, the FIESTA-IoT project<sup>37</sup> gives access to federated datasets described with a unique ontology. On the one hand, this approach ensures complete semantic interoperability, but on the other hand, it restricts the expressivity available for data providers: datasets already described with another vocabulary need to be transformed to be compliant with the fiesta ontology<sup>38</sup>. The transformation can be manual, but it can also be systematized using automatic ontology alignments approaches. However, automatic ontologies matching techniques still need evaluation on IoT ontologies. If simple alignments are not sufficient, complex alignments should also be considered.

Moreover, being associated to M2M communication only, IoT content is mostly numeric, and has to be interpreted by applications when displayed to a user. The enrichment of content with ontologies enables new approaches to user interaction, based on natural language for both user requirements expression, and query answering. This perspective would make IoT, WoT and SWoT applications easier to access for non-expert users, bringing services to humans at the core of nodes networks.

---

<sup>37</sup><http://fiesta-iot.eu/>

<sup>38</sup><http://ontology.fiesta-iot.eu>

## References

- [1] I. Szilagyí and P. Wira, "Ontologies and Semantic Web for the Internet of Things - a survey," in *IECON*, IEEE, 2016.
- [2] ITU, "Framework of the web of things," tech. rep., International Telecommunication Union.
- [3] J. Rivera and R. van der Meulen, "Gartner says the internet of things installed base will grow to 26 billion units by 2020," tech. rep., Gartner cabinet.
- [4] P. Desai, A. Sheth, and P. Anantharam, "Semantic Gateway as a Service architecture for IoT Interoperability," in *Kno.e.sis Publications*, 2015.
- [5] A. Gyrard, M. Serrano, and G. A. Ateamezing, "Semantic web methodologies, best practices and ontology engineering applied to Internet of Things," in *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, pp. 412–417, IEEE, 2015.
- [6] F. Aïssaoui, G. Garzone, and N. Seydoux, "Providing interoperability for autonomic control of connected devices," in *InterIoT*, pp. 1–7, 2016.
- [7] I. Skerrett, "IoT Developer Trends 2017 Edition," 2017.
- [8] P. Murdock and al., "Semantic Interoperability for the Web of Things," 2016.
- [9] O. Corcho and R. García-Castro, "Five challenges for the Semantic Sensor Web," *Semantic Web*, vol. 1, no. 1, pp. 121–125, 2010.
- [10] M. Ben-Alaya, S. Medjah, T. Monteil, and K. Drira, "Toward semantic interoperability in oneM2M architecture," *IEEE Communications Magazine*, vol. 53, no. 12, pp. 35–41, 2015.
- [11] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the internet of things: A survey," *IEEE Communications Surveys and Tutorials*, vol. 16, pp. 414–454, jan 2014.
- [12] D. Pfisterer, K. Romer, D. Bimschas, O. Kleine, R. Mietz, C. Truong, H. Hasemann, A. Kröller, M. Pagel, M. Hauswirth, M. Karnstedt, M. Leggieri, A. Passant, and R. Richardson, "SPITFIRE: toward a semantic web of things," *IEEE Communications Magazine*, vol. 49, pp. 40–48, nov 2011.
- [13] V. Foteinos, D. Kelaidonis, G. Poullos, P. Vlacheas, V. Stavroulaki, and P. Demestichas, "Cognitive management for the internet of things: A framework for enabling autonomous applications," *IEEE Vehicular Technology Magazine*, vol. 8, no. 4, pp. 90–99, 2013.
- [14] M. G. Kibria, "Knowledge based open IoT service provisioning through cooperation between physical web and WoO," in *2015 Seventh International Conference on Ubiquitous and Future Networks*, pp. 395–400, IEEE, 2015.
- [15] M. Mrissa, L. Medini, J.-P. Jamont, N. Le Sommer, and J. Laplace, "An Avatar Architecture for the Web of Things," *Internet Computing, IEEE*, vol. 19, no. 2, pp. 30–38, 2015.
- [16] M. Eid, R. Liscano, and A. El Saddik, "A universal ontology for sensor networks data," *Proceedings of the 2007 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, CIMSAS*, no. JULY, pp. 59–62, 2007.
- [17] L. Lefort and C. Henson, "Semantic sensor network xg final report," Tech. Rep. June 2011, 2011.
- [18] N. Seydoux, K. Drira, N. Hernandez, and T. Monteil, "IoT-O, a core-domain IoT ontology to represent connected devices networks," in *EKAW*, 2016.
- [19] A. Zanello, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for Smart Cities," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, 2014.
- [20] D. Le-phuoc and M. Hauswirth, "Linked open data in sensor data mashups," in *Proceedings of the 2nd International Workshop on Semantic Sensor Networks (SSN09)*, vol. 522, pp. 1–16, 2009.
- [21] J. Liu, Y. Li, M. Chen, W. Dong, and D. Jin, "Software-defined internet of things for smart urban sensing," *IEEE Communications Magazine*, vol. 53, no. 9, pp. 55–63, 2015.
- [22] M. Compton, C. Henson, L. Lefort, H. Neuhaus, and A. Sheth, "A survey of the semantic specification of sensors," 2009.
- [23] P. Barnaghi, S. Meissner, M. Presser, and K. Moessner, "Sense and sens' ability: Semantic data modelling for sensor networks," 2009.
- [24] S. Nikoli, V. Penca, and Z. Konjovi, "Semantic Web Based Architecture for Managing Hardware Heterogeneity in Wireless Sensor Network," in *International Journal of Computer Science and Applications*, vol. 8, pp. 38–58, 2011.
- [25] P. Vlacheas, R. Giffreda, V. Stavroulaki, D. Kelaidonis, V. Foteinos, G. Poullos, P. Demestichas, A. Somov, A. Biswas, and K. Moessner, "Enabling smart cities through a cognitive management framework for the internet of things," *IEEE Communications Magazine*, vol. 51, no. 6, pp. 102–111, 2013.
- [26] H. Hasemann, A. Kröller, and M. Pagel, "RDF provisioning for the internet of things," in *Proceedings of 2012 International Conference on the Internet of Things, IOT 2012*, pp. 143–150, IEEE, oct 2012.
- [27] J. Rowley, "The wisdom hierarchy: representations of the DIKW hierarchy," *Journal of Information Science*, vol. 33, no. 2, pp. 163–180, 2007.
- [28] M. Ma, P. Wang, and C.-H. H. Chu, "Ontology-Based Semantic Modeling and Evaluation for Internet of Things Applications," in *2014 IEEE International Conference on Internet of Things (iThings), and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom)*, no. iThings, pp. 24–30, IEEE, 2014.
- [29] A. Sheth, C. Henson, and S. S. Sahoo, "Semantic Sensor Web," in *IEEE Internet Computing*, vol. 12, pp. 78–83, 2008.
- [30] C. Henson, A. Sheth, and K. Thirunarayan, "Semantic perception: Converting sensory observations to abstractions," *IEEE Internet Computing*, vol. 16, no. 2, pp. 26–34, 2012.
- [31] S. Poslad, S. E. Middleton, F. Chaves, R. Tao, O. Necmioglu, and U. Bugel, "A Semantic IoT Early Warning System for Natural Environment Crisis Management," *IEEE Transactions on Emerging Topics in Computing*, vol. 3, no. 2, pp. 246–257, 2015.
- [32] Z. Ming, F. Hong, and M. Yan, "Semantic annotation method of IOT middleware," in *2013 Fourth International Conference on Intelligent Control and Information Processing (ICICIP)*, pp. 495–498, IEEE, 2013.
- [33] D. Le-Phuoc, H. Nguyen Mau Quoc, H. Ngo Quoc, T. Tran Nhat, and M. Hauswirth, "The Graph of Things: A step towards the Live Knowledge Graph of connected things," *Web Semantics: Science, Services and Agents on the World Wide Web*, 2016.
- [34] M. Ruta, S. Ieva, G. Loseto, and E. Di Sciascio, "From the Physical Web to the Physical Semantic Web: knowledge discovery in the Internet of Things," in *enth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*, 2016.
- [35] L. Daniele, M. Solanki, F. D. Hartog, and J. Roes, "Interoperability for Smart Appliances in the IoT World," in *ISWC*, 2016.

- [36] M. Bermudez-Edo, T. Elsaiah, P. Barnaghi, and K. Taylor, "IoT-Lite: A Lightweight Semantic Model for the Internet of Things," in *IEEE International conference on Ubiquitous Intelligence and computing*, no. November, 2016.
- [37] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Computer Networks*, vol. 54, pp. 2787–2805, oct 2010.
- [38] P. Barnaghi, W. Wang, C. Henson, and K. Taylor, "Semantics for the Internet of Things: early progress and back to the future," *International Journal on Semantic Web and Information Systems*, vol. 8, no. 1, pp. 1–21, 2012.
- [39] A. J. Jara, A. C. Olivieri, Y. Bocchi, M. Jung, W. Kastner, and A. F. Skarmeta, "Semantic Web of Things: an analysis of the application semantics for the IoT moving towards the IoT convergence," *Int. J. of Web and Grid Services*, vol. 10, no. 2/3, 2014.
- [40] J. Kiljander, A. D'elia, F. Morandi, P. Hyttinen, J. Takalo-Mattila, A. Ylisaukko-Oja, J.-P. Soininen, and T. S. Cinotti, "Semantic Interoperability Architecture for Pervasive Computing and Internet of Things," *IEEE Access*, vol. 2, pp. 856–873, 2014.
- [41] J. Kopecký, T. Vitvar, C. Bournez, and J. Farrell, "SAWSDL: Semantic Annotations for WSDL and XML Schema," *IEEE Internet Computing*, vol. 11, pp. 60–67, nov 2007.
- [42] Y. Qin, Q. Z. Sheng, and E. Curry, "Matching Over Linked Data Streams in the Internet of Things," *IEEE Internet Computing*, vol. 19, no. 3, pp. 21–27, 2015.
- [43] J.-p. J. Calbimonte, H. Jeung, O. Corcho, and K. Aberer, "Semantic Sensor Data Search in a Large-Scale Federated Sensor Network," *Semantic Sensor Networks*, pp. 14–29, 2011.
- [44] I. Chatzigiannakis, H. Hasemann, M. Karnstedt, O. Kleine, A. Kröller, M. Leggieri, D. Pfisterer, K. Römer, and C. Truong, "True Self-Configuration for the IoT," in *3rd International Conference on the Internet of Things (IOT)*, 2012.
- [45] M. I. Ashraf, L. Goratti, J. Haapola, and C. Pomalaza-Raez, "Distributed semantic algorithm for power constrained publish/subscribe routing," in *2010 Sixth International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pp. 91–96, IEEE, dec 2010.
- [46] S. B. Fredj, M. Boussard, D. Kofman, and L. Noirie, "A Scalable IoT Service Search Based on Clustering and Aggregation," in *2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing*, pp. 403–410, IEEE, aug 2013.
- [47] F. Bouhafs, M. Merabti, and H. Mokhtar, "A semantic clustering routing protocol for wireless sensor networks," in *CCNC 2006 3rd IEEE Consumer Communications and Networking Conference, 2006.*, vol. 1, pp. 351–355, IEEE, 2006.
- [48] E. Siow, T. Tiropanis, and W. Hall, "SPARQL-to-SQL on Internet of Things Databases and Streams," in *ISWC*, 2016.
- [49] Y. Qin, Q. Z. Sheng, N. J. Falkner, S. Dustdar, H. Wang, and A. V. Vasilakos, "When Things Matter: A Survey on Data-Centric Internet of Things," *Journal of Network and Computer Applications*, no. April, 2016.
- [50] Z. Li, C. H. Chu, W. Yao, and R. a. Behr, "Ontology-driven event detection and indexing in smart spaces," in *Proceedings - 2010 IEEE 4th International Conference on Semantic Computing, ICSC 2010*, pp. 285–292, 2010.
- [51] D. Hussein, S. N. Han, G. M. Lee, N. Crespi, and E. Bertin, "Towards a dynamic discovery of smart services in the social internet of things," *Computers & Electrical Engineering*, 2016.
- [52] S. K. Datta and C. Bonnet, "Connect and Control Things : Integrating Lightweight IoT Framework into a Mobile Application," in *9th International Conference on Next Generation Mobile Applications, Services and Technologies*, no. SEPTEMBER 2015, p. 6, 2015.
- [53] Z. Wang, R. Bie, and M. Zhou, "Hybrid ontology matching for solving the heterogeneous problem of the IoT," *Proc. of the 11th IEEE Int. Conference on Trust, Security and Privacy in Computing and Communications, TrustCom-2012 - 11th IEEE Int. Conference on Ubiquitous Computing and Communications, IUCC-2012*, pp. 1799–1804, 2012.
- [54] D. Le-Phuoc, H. Q. Nguyen-Mau, J. X. Parreira, and M. Hauswirth, "A middleware framework for scalable management of linked streams," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 16, pp. 42–51, 2012.
- [55] V. Charpenay, S. Kabisch, D. Anicic, and H. Kosch, "An ontology design pattern for IoT device tagging systems," in *Proceedings - 2015 5th International Conference on the Internet of Things, IoT 2015*, 2015.
- [56] C. Sarkar, A. U. Akshay, R. V. Prasad, A. Rahim, R. Neisse, and G. Baldini, "DIAT: A scalable distributed architecture for IoT," in *IEEE Internet of Things Journal*, vol. 2, 2015.
- [57] S. Nambi, C. Sarkar, R. V. Prasad, and a. Rahim, "A unified semantic knowledge base for IoT," *Internet of Things (WF-IoT), 2014 IEEE World Forum on*, pp. 575–580, 2014.
- [58] S. De, P. Barnaghi, M. Bauer, and S. Meissner, "Service modelling for the Internet of Things," in *2011 Federated Conference on Computer Science and Information Systems (FedCIS)*, pp. 949–955, 2011.
- [59] J. Li, N. Zaman, and H. Li, "A Decentralized Locality-Preserving Context-Aware Service Discovery Framework for Internet of Things," in *2015 IEEE International Conference on Services Computing*, pp. 317–323, IEEE, jun 2015.
- [60] D. J. Russomanno, C. Kothari, and O. Thomas, "Sensor ontologies: From shallow to deep models," in *Annual Southeastern Symposium on System Theory*, vol. 37, pp. 107–112, 2005.
- [61] C. Perera, A. Zaslavsky, C. H. Liu, M. Compton, P. Christen, and D. Georgakopoulos, "Sensor search techniques for sensing as a service architecture for the internet of things," *IEEE Sensors Journal*, vol. 14, no. 2, pp. 406–420, 2014.
- [62] S. N. Han, G. M. Lee, and N. Crespi, "Semantic Context-Aware Service Composition for Building Automation System," *IEEE Transactions on Industrial Informatics*, vol. 10, pp. 752–761, feb 2014.
- [63] A. I. Maarala, X. Su, and J. Riekk, "Semantic data provisioning and reasoning for the Internet of Things," in *2014 International Conference on the Internet of Things, IOT 2014*, 2014.
- [64] F. Ganz, D. Puschmann, P. Barnaghi, and F. Carrez, "A Practical Evaluation of Information Processing and Abstraction Techniques for the Internet of Things," *IEEE Internet of Things Journal*, vol. 2, no. 4, pp. 340–354, 2015.
- [65] K. Kotis, A. Katasonov, and J. Leino, "Aligning Smart and Control Entities in the IoT," in *Internet of Things, Smart Spaces, and Next Generation Networking: 12th International Conference, NEW2AN 2012, and 5th Conference, ruSMART 2012, St. Petersburg, Russia, August 27-29, 2012. Proceedings* (S. Andreev, S. Balandin, and Y. Koucheryavy, eds.), vol. 7469 LNCS, pp. 39–50, Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.
- [66] W. Wang, S. De, R. Toenjes, E. Reetz, and K. Moessner, "A

- comprehensive ontology for knowledge representation in the internet of things,” *Proc. of the 11th IEEE Int. Conference on Trust, Security and Privacy in Computing and Communications, TrustCom-2012 - 11th IEEE Int. Conference on Ubiquitous Computing and Communications, IUCC-2012*, pp. 1793–1798, 2012.
- [67] F. Ganz, P. Barnaghi, and F. Carrez, “Automated Semantic Knowledge Acquisition From Sensor Data,” *Systems Journal, IEEE*, vol. PP, no. 99, pp. 1–12, 2014.
- [68] E. Kharlamov, Y. Kotidis, T. Mailis, C. Neuenstadt, C. Nikolaou, Ö. Özcep, C. Svingos, D. Zheleznyakov, S. Lamparter, I. Horrocks, Y. Ioannidis, and R. Möller, “Towards Analytics Aware Ontology Based Access to Static and Streaming Data (Extended Version),” in *ISWC*, pp. 1–22, 2016.
- [69] Q. Wu, G. Ding, Y. Xu, S. Feng, Z. Du, J. Wang, and K. Long, “Cognitive Internet of Things: A New Paradigm Beyond Connection,” *IEEE Internet of Things Journal*, vol. 1, pp. 129–143, apr 2014.
- [70] L. Nachabe, M. Girod-Genet, and B. El Hassan, “Semantic Techniques for IOT Data and Service Management on to Smart System,” *International Journal of Wireless & Mobile Networks*, vol. 8, no. 4, pp. 43–63, 2016.
- [71] M. Compton, H. Neuhaus, K. Taylor, and K. Tran, “Reasoning about Sensors and Compositions,” 2009.
- [72] X. Su, J. Riekkki, J. K. Nurminen, J. Nieminen, and M. Koskimies, “Adding semantics to internet of things,” *Concurrency Computation*, vol. 27, no. 8, 2015.
- [73] G. Loseto, S. Ieva, F. Gramegna, M. Ruta, F. Scioscia, E. D. Sciascio, and B. I., “Linked Data ( in low-resource ) Platforms : a mapping for Constrained Application Protocol,” in *ISWC*, (Kobe), 2016.
- [74] H. R. Bazoobandi, S. De Rooij, J. Urbani, A. Ten Teije, F. Van Harmelen, and H. Bal, “A Compact In-Memory Dictionary for RDF Data,” in *ESWC*, vol. 9088, pp. 205–220, 2015.
- [75] J.-P. Calbimonte, H. Jeung, O. Corcho, and K. Aberer, “Enabling Query Technologies for the Semantic Sensor Web,” *International Journal on Semantic Web and Information Systems*, vol. 8, no. 1, pp. 43–63, 2012.
- [76] C. A. Lana, N. M. Souza, M. E. Delamaro, E. Y. Nakagawa, F. Oquendo, and J. C. Maldonado, “Systems-of-systems development: Initiatives, trends, and challenges,” in *2016 XLII Latin American Computing Conference (CLEI)*, pp. 1–12, IEEE, oct 2016.