AN ANALYSIS OF CONTEXT-AWARE DATA MODELS FOR SMART CITIES: TOWARDS FIWARE AND ETSI CIM EMERGING DATA MODEL

Antonio J. Jara[a],[*] Yann Bocchi[a], David Fernandez[b], German Molina[b], Andrea Gomez[b,]

[a] University of Applied Sciences Western Switzerland (HES-SO), Technopole, 3, 3960 Sierre Switzerland - {antonio.jara, yann.bocchi}@hevs.ch
[b] HOP Ubiquitous S.L. (HOPU), Murcia, Spain - {german, david, andrea}@hopu.eu
[c] Universidad Catolica de San Antonio (UCAM), Murcia, Spain

KEY WORDS: FIWARE, Context Awareness, ETSI CIM, Data Models, Smart Cities, Smart Destinations

ABSTRACT:

Smart Cities requires the support of context-aware and enriched semantic descriptions to support a scalable and cross-domain development of smart applications. For example, nowadays general purpose sensors such as crowd monitoring (counting people in an area), environmental information (pollution, air quality, temperature, humidity, noise) etc. can be used in multiple solutions with different objectives. For that reason, a data model that offers advanced capabilities for the description of context is required. This paper presents an overview of the available technologies for this purpose and how it is being addressed by the Open and Agile Smart Cities principles and FIWARE platform through the data models defined by the ETSI ISG Context Information Management (ETSI CIM).

1. INTRODUCTION

Smart Cities require IoT knowledge-driven and context-awareness solutions, ubiquitous services, and smart applications development. Figure 1 shows the phases that IoT has followed during the last decade to become a semantic IoT powered by Web technologies.

The first phase was to interconnect everything to Internet. The second phase enabled a seamless interoperability among the heterogeneous entities.The existing heterogeneous islands of devices were interconnected using IPv6. The integration was established at the connectivity level with solution such as 6LoWPAN (Hui and Thubert (2011)) and GLoWBAL IPv6 (Jara et al. (2012a)).

Once that the connectivity was reached, IoT needed a common protocol for the transport layer to connect things to the Web. For that reason, the next phase was connect things to the Web using the standard solutions already adopted in the Web (HTTP, HTML, etc.), thereby conceiving the so-called Web of Things.

Protocols from the Internet Engineering Task Force (IETF) such as Constrained Application Protocol (CoAP) (Shelby et al. (2013)), which is mapped to HTTP, and also lightweight implementations of HTTP make the interaction with resources from constrained devices feasible, through browsers and with the simplicity and flexibility that the Web offers nowadays.

Web of Things allows the different things and systems to interact together, thereby allowing to compose more complex services and solutions. These interactions are enabled through the definition of application programming interfaces (API) over HTTP or CoAP protocol. Thereby, the applications leverage the HTTP protocol to provide the interface for publishing data updates into the system, for retrieving data updates from the system, and in general exchange of information.

The data can be encoded with different envelopes, semantics and metadata. Here I bring to mind that the same data can be gathered
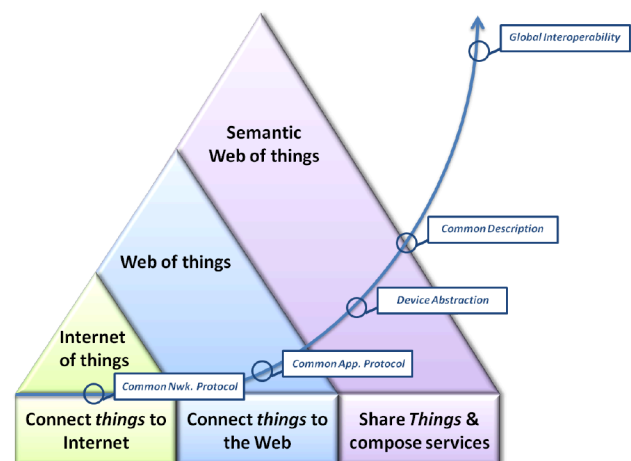
---
*Corresponding author

Figure 1. Evolution of the market size from the Internet of Things to the semantic Web of Things.

in plain text or encapsulated over complex structures such as XML/EXI or simpler but yet organized structures such as JSON. In addition, they can be represented with different format and units, and finally they can offer additional information.

The initial market of the IoT has been focused on deployments that are connected vertically, with specific sensors and applications for which they have been designed in order to address specific requirements and target a specific use case. An example is smart metering with specific protocols for this purpose such as DLMS; and platforms for energy management etc., at the same way, we can find similar solutions for water management with protocols such as WITS. However, these vertical solutions are not longer feasible to keep it isolated. smart Cities are requiring a major level of integration and interoperability, and for that, semantics are the major driver.

The challenge after the Web of Things, is to build a semantic Web of Things (SWoT) in order to ensure a common understanding.

The SWoT is, on the one hand, the fusion of the trends of the IoT

for moving toward the Web technologies with protocols such as CoAP, REST architecture and the Web of Things concept, and on the other hand, the evolution of the Web with the Semantic Web technologies and the capacity to provide context information to the collected data.

SWoT promises a seamless extension to the IoT allowing integration of both the physical and digital worlds. SWoT is focused on providing wide scale interoperability that allows the sharing and re-use of these things. Thereby, the use cases and markets of the IoT will not be held back to vertical solutions or pre-established use cases. In fact, these deployed infrastructures and available data can target other secondary markets and use cases. In fact the data they collect and manage can be useful for data analysis (aggregated, anonymity, processed information, e.g., for Smart City administration), and even provide a major understanding for the primary markets, since they can be compared and extended with the available third party data.

Therefore, moving from the IoT/WoT towards the SWoT is challenging. Some challenges are to define a common description that allows data to be universally understandable, create extensible annotations, i.e., from minimal semantic descriptions towards more elaborate ones, and agree on a catalogue of semantic descriptions; i.e. data models that can be used for the Smart Cities platforms, dashboards and different Smart Applications.

First, the standards considered for cellular networks have been initialized by the European side with the ETSI M2M and extended globally with the oneM2M initiative, which is already offering the OMA Lightweight Device Management Protocol (OMA LwM2M); in conjunction with other protocols already consolidated in the market; at the same time that provides all the capabilities for scalable management and interfacing with third party platforms.

Second, the standards considered for capillary networks are supported by organizations such as the Internet Engineering Task Force with solutions such as CoAP, which is supported by industry alliances such as IPSO Alliance, with the IPSO Application Profile. The capillary networks present the major heterogeneity and other standards for offering a lightweight reliable messaging transport protocol for the IoT such as the Message Queuing Telemetry Transport (MQTT) protocol optimized to connect physical world devices and events with enterprise servers and other consumers supported by OASIS and Eclipse Foundation (Davis et al. (2013)).

Other activities and projects are the W3C with the SSN-XG ontology for offering a semantic layer for the IoT, the European Research Cluster such as AIOTI. These different approaches are described in the Section 2.

For this purpose, we carry out an evaluation of the ongoing works and we research in the scientific and industrial community to develop the semantic-layer for the IoT with a comprehensive analysis of each technology. It begins, on the one hand, from the qualitative point of view with a comparison about its potential, functionality, and facilities for knowledge engineering, and on the other hand, from the quantitative point of view an analysis about the performance, overhead, and footprint. This analysis is presented in the Section 3, with a discussion about the trends and vision in the Section 4. Finally, the Section 5 concludes this paper.
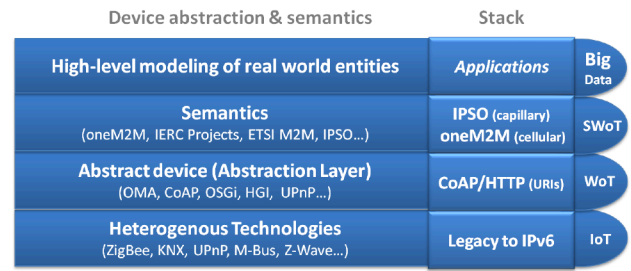

Figure 2. Device abstractions and semantics to enable the Semantic Internet of Things.

1.1 Interfacing with heterogeneous resources

SWoT aims to integrate information which is semantically rich and easily accessible into the physical world, thus connecting smart objects and digital entities. The SWoT will impact on the cyber-physical systems integration and the human machine interaction models.

Until now the IoT has promoted the global scale integration of the identification, location and data coming from everywhere and everything. The SWoT vision enables knowledge-based systems that achieve high degrees of autonomic capability for information storage, management and discovery, and providing transparent access to information sources in a given area.

A very related word with the IoT is constrained, in terms of a small amount of information due to memory capacity, low power and cost features. Devices are usually equipped with little processing capabilities, short-range coverage and low-throughput wireless links allowing only a simple service/resource functionality.

Figure 2 presents the stack to enable the SWoT. This presents a bottom-up approach which covers the integration of heterogeneous technologies to the support the development of applications providing a high-level modeling of real world entities with semantic information.

The IoT requires the integration of a wide range of technologies, from legacy technologies of home and building automation such as BACnet, KNX and Z-Wave and emerging technologies for smart grid and smart cities, to any class of sensor, actuator, tag or thing.

For the support of the heterogeneity the first step for the IoT, such as presented at the bottom of the Figure 2, has been the mapping and the integration of everything to a common addressing space, i.e., IPv6. Thereby, allowing to connect all the things in a homogeneous way.

The IPv6 integration is being reached through 6LoWPAN (Hui and Thubert (2011)) and GLoWBAL IPv6 (Jara et al. (2012a)) for emerging IoT resources, but also through IPv6 Addressing Proxies for legacy technologies such as those described from the home and building automation sectors (Jara et al. (2013)).

In case where IPv6 is not integrated, the other integration is through a gateway or a middleware, but these approaches break the end-to-end connectivity with the resources, which happens to be one of the main principles of the IoT.

1.2 Device abstraction

Related with the heterogeneous device integration, a device abstraction layer is required in order to access all the resources through a common protocol and representation.

Device abstraction in IoT is enabled by Web technologies. Specifically, the Representation State Transfer (RESTFul) architecture style can be used. RESTFul is defining one of the most powerful mechanisms over the World Wide Web to build communication interfaces and protocols that enable the exchange of information and interoperability among systems.

CoAP is an open standard to build embedded RESTful Web services optimized for constrained devices such as that located in the IoT deployments. CoAP has been designed by the Internet Engineering Task Force (IETF), and nowadays it is supported and used by the IPSO Alliance for the definition of the IPSO Application Profile (Shelby and Chauvenet (2012)) and the Lightweight OMA Device Management (DM) (Chu et al. (2013)).

RESTFul offers a resource facing architecture where all the resources are identified through a Universal Resource Identifier (URI) (Berners-Lee et al. (2005)).

The URIs allow a uniform device exposure and uniform resource access. Therefore, all the devices can be abstracted to URIs.

The description over the URI, i.e., the semantic abstraction is being defined by multiple standardization organizations, such as ETSI M2M, Home Gateway Initiative (HGI), OSGi, OMA, and IPSO Application Framework.

Therefore, collaboration on standardizing a coherent solution in terms of abstraction semantics to overcome this barrier of a standardized heterogeneity is required.

1.3 Syntactic and Semantic interoperability

The syntactic interoperability is the first challenge, since the Web is supporting multiple content types, some of the most common representations of the resources can be text plain, JSON or XML.

JSON is gaining special attention in the IoT market, since it is lightweight, simple and offers capabilities close to the XML ones without requiring the overhead (e.g., schema) and processing requirements of XML.

Once the resource is encapsulated over a supported content type, the resource should be properly described in the semantic-enabled Web. For example, IoT has defined the Web Linking in order to build semantically annotated resource descriptions.

Therefore, Semantic Web of Things is offering a common protocol (HTTP/CoAP), common methods (GET, PUT, POST and DELETE from HTTP/CoAP), universal identifiers (URIs), and a common solution to annotate resources (Web Linking), required to put together a common reference (i.e., an ontology).

The common reference is crucial for the SWoT in order to enable business services with machine interpretable descriptions.

This common semantic reference will allow the service composition to offer services with higher context awareness and knowledge, the re-use of service components, the capability to abstract complex and heterogeneous platform in large scale deployments,

the context-aware service adaptation, and finally even more facility for validation, testing, discoverability, reasoning and decision making.

The analysis in details of the semantic trends is described in the next section. In addition, it is evaluated in the Section 3 the capabilities of the different standards and performance of the different resources representation.

2. SEMANTIC SERVICES DESCRIPTION AND STANDARDS FOR THE IOT

The next subsections analyse the different protocols and standards.

2.1 CoAP and IPSO Application Framework

The main goal of the Constrained RESTFul Environments (CoRE) working group in the IETF has been the development of a protocol that follows up the REST architecture guidelines and fits with the constrained nodes and networks capabilities.

CoAP is the protocol proposed by CoRE for resource access and transport, that satisfies the required functionality of the IoT transmission technologies (Shelby et al. (2013)).

CoAP offers a wrapper for transport of the data similar to HTTP, but optimized for bandwidth and frame size constrained devices.

CoRE has also been defined for CoAP a resource directory, CoAP observe, CoAP block-wise, and Web Linking for discovery and binding.

CoRE Resource Directory allocates all resources and services offered by a device, making them discoverable either through a direct link in */.well-known/core* or by following successive links starting from */.well-known/core*, defined in the Web Linking format.

CoAP observe and the conditional observe offer a mechanism for subscribing to changes on the sensor under specific conditions. CoAP conditional observe is one of our previous works located at (Jara et al. (2012b); Ruta et al. (2007)).

CoAP is a very well designed protocol with several ancillary protocols that offers a very powerful mechanism for the development of the IoT, but CoAP does not define the content, therefore it is applicable to higher level standards and application frameworks in order to solve the needs from the real world applications.

Specifically, it is being used by the IPSO Application Framework from the IPSO Alliance. It defines RESTFul interfaces for the definition and management of resource lists, batch, sensors, parameter, actuators and binding tables of resources. For example, the semantic IPSO Application Framework has chosen SenML over JSON with the usage of the Unified Code for Units of Measure (UCUM). These initial semantic capabilities allow avoiding the initial mistakes from CoAP such as the use of inappropriate unit codes such as *23 C* for temperature, when it is according to the UCUM standard means velocity of light, and consequently this should be *23 CEL*.

The current semantic capabilities from IPSO Application Framework are very basic in order to offer a very simple and lightweight solution.

2.2 Lightweight OMA Device Management and oneM2M

Lightweight OMA Device Management is a protocol for device management, the use this protocol in M2M requires efficient message formats and transport replacement such as CoAP, and Core Link Format. For that reason, Lightweight OMA DM has chosen CoAP to provide the core functionalities of HTTP (GET, PUT, POST, DELETE commands) in a reduced footprint.

In addition, it focuses on providing mechanisms for asynchronous and synchronous communication, store, forward and caching mechanism for optimizing the communication, and security with mechanisms to provide two way authentication and secure communication channels.

Lightweight OMA DM is supported by the oneM2M, which provides an international initiative that will play a very relevant role to propose the standards for the syntactic and semantic information.

oneM2M will define abstraction layers, using the same format. This will ease the creation of the higher-layers for the IoT and M2M that enables a high-level modeling of real world entities, development of applications, and finally huge quantities of data collection, such as presented in the top layer of the Figure 2.

oneM2M will also offer support and solutions to facilitate the development of vertical industries and new markets.

Finally, once an agreement on the abstraction layer and semantics has been achieved, including references to the semantic descriptions in oneM2M specifications (to enable machine interpretation), it will be coordinated with the other described institutions such as HGI, Broadband Forum, OSGi, etc. Note, that some of the existing institutions such as ZigBee Alliance are already part of the oneM2M initiative.

2.3 ETSI M2M

ETSI M2M is a service-oriented architecture to build the Service Capabilities Layer (SCL) for M2M/IoT devices, M2M/IoT Gateways, and M2M/IoT Servers.

ETSI M2M standardizes the resource tree structure that resides on the M2M SCL from each one of the components. These components exchange information by means the standards-based reference points. The reference points enable the interoperability between the mentioned components, i.e., devices, gateways and servers. Specifically, they are defined the denominated dIa/mId/mIa reference points (ETSI (n.d.)).

ETSI M2M interfaces are being implemented following the REST-Ful architectures style over HTTP and CoAP protocols. The information is represented by a tree of resources, that uses XML-based or JSON-based representations for information interchange.

The dIa interface between the devices and the gateways (a.k.a. M2M Gateway Service Capability - GSCL), the mId interface between the gateways and the servers (a.k.a. M2M Network Service Capabilities Layer - M2M NSCL), and the mIa interface between the M2M NSCL and the network applications.

These interfaces provide the functionality for the registration of devices/gateways to the backend, request to get the authorization to read or write a resource, subscription and notifications for specific events, and device management operations.

In addition to the interfaces, ETSI M2M also offers the identification of the application and devices requirements for asynchronous and synchronous communications, quality o service mechanisms based on policies for optimizing the communication, and security for mutual authentication between M2M NSCL and device/gateways and secure channel establishment for data transportation.

ETSI M2M is re-using existing and well-defined standards for the device management. On the one hand, device management based both on OMA DM for wireless communications, i.e., the protocol also considered for the oneM2M, and on the other hand, BBF TR-69 from the broadband forum for wireline communications.

Finally, ETSI M2M implementations are being developed by projects such as FI-WARE which has developed preliminary instances of the M2M interface (FI-WARE (2012)), and by companies such as Radisys, Grid2Home, Intecs, Intel, InterDigital, Sensinode and Telecom Italia (Interdigital (2012)). They have tested several types of devices for different applications, and the integration with technologies such as ZigBee, WiFi and cellular (GPRS and UMTS).

2.4 ETSI ISG CIM

ETSI ISG Context Information Management (ETSI ISG CIM). FIWARE initiative and platform through the Orion Context Broker component identified a key market need for IoT and Smart Cities; it is the management of context in a scalable and standardized way. For this purpose, FIWARE defined on the one hand, OMA NGSI interfaces to offer a homogeneous access to data, and on the other hand, a set of data models being standardized by ETSI ISG CIM.

Context Information provides the meta-data structure for sensors measurement and also other data feeds from video, social media etc. Even when context is very simple to understand by human being, in order to provide artificial intelligence capabilities to smart systems, it is crucial to formalize and provide much more details about the context and make it available in conjunction with the data. A Context Information Management (CIM) system acts as a clearing-house for publishing, discovering, monitoring and maintaining data according to relevant contexts for smart applications.

"ETSI ISG CIM will specify protocols running on top of IoT platforms and allowing exchange of data together with its context, this includes what is described by the data, what was measured, when, where, by what, the time of validity, ownership, and others. That will dramatically extend the interoperability of applications, helping smart cities to integrate their existing services and enable new third-party services", as stated by the ETSI ISG CIM convenor, Lindsay Frost.

ETSI ISG CIM will focus on developing specifications for a common context information management API, data publication platforms and standard data models. A practical example of these data models can be found in FIWARE data models: `https://www.fiware.org/data-models/`.

Figure 3 presents an overview of the available data models defined for Smart Cities by ETSI ISG CIM. More practical examples about these data models for Smart Points of Interactions (Smart POIs), security, healthcare, air quality monitoring and smart destinations are being described in a joint action between Europe and Mexico in SmartSDK ((SmartSDK, 2011; D. Fernandez, 2017)).
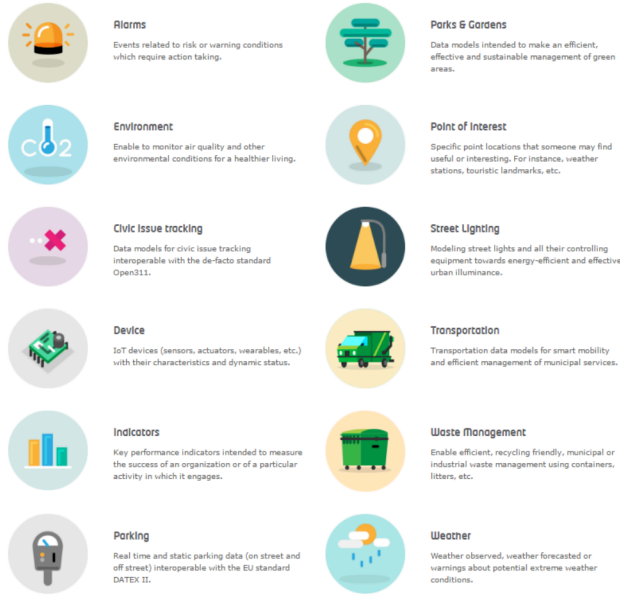
Figure 3. FIWARE data models for Smart Cities based on ETSI ISG CIM.

2.5 Ontology-based Resource Description and Discovery Framework

Semantics is the idea of giving meaning to things, whatever they are. But if we want to apply this idea in the context of the IoT, being within the Web, we need the semantic Web. It has the same intents of semantics, but with some constraints. In fact when we talk about the semantic Web we do not talk just about giving semantic representation to the sensors, the actuators and to the other entities involved in the IoT, but we must focus our attention also on the fact that this semantic representation and the data attached to it have to be passed around in the World Wide Web to allow software to store, exchange and use machine-readable information distributed throughout the Web. The Semantic Web was built over other W3C standards, the RDF data model, the RDF Schema and OWL standards and the SPARQL query language and if a solution uses semantics, but it does not use these standards, it cannot be part of the semantic Web.

The Resource Description Framework (RDF) is the specification of W3C to design the description and to model information for the Web resources. The RDF data model is based on a basic unit of information called triple formed by subject-predicate-object relationship. A subject denotes a resource, a predicate denotes the relationship between the subject and an object, and an object denotes the value of the related subject.

RDF uses URIs to represent subjects and predicates in order to assure the unambiguity of information in the whole Web. A collection of RDF statements represents a graph which provides knowledge representation, that can be persisted using relational databases or triple stores.

RDF Schema (RDFS) is the Vocabulary Description Language for RDF and it is composed of a set of classes with certain properties using RDF. It is the base for the more expressive language OWL (Web Ontology Language).

The OWL Web Ontology Language is used to model knowledge bases. OWL is the practical realization of a description logic known as $\mathcal{SHOIN}(\mathcal{D})$(Horrocks et al. (2003)). OWL is aligned to RDF and it defines classes (also called concepts in the DL literature), properties, and individuals that can be compared to subjects, predicates and objects in RDF. An OWL ontology consists of a set of class axioms that point to logical relationships between classes, which constitutes a TBox (Terminological Box); a set of property axioms to specify logical relationships between properties, which constitutes a RBox (Role Box); and a collection of assertions that describe individuals, which constitutes an ABox (Assertional Box). Classes are formal descriptions of sets of objects (taken from a non empty universe), and individuals which represent the names of objects of the universe. Properties can be either object properties, which represent binary relations between objects of the universe, or data properties, which represent binary relationships between objects and data values (taken from XML Schema datatypes). Class axioms allow one to specify which subclass ($\sqsubseteq$) or equivalence ($\equiv$) relationships exist between certain classes and the domain and range of a property. Assertions allow one to specify that an individual belongs to a class: $C(a)$ means that the object denoted by a belongs to the class C; and that an individual is related to another individual through an object property: $R(b,c)$ means the object denoted by b is related to the object denoted by c through the property R. Complex classes can be specified by using Boolean operations on classes: $C \sqcup D$ is the union of classes, $C \sqcap D$ is the intersection of classes, and $\neg C$ is the complement of class C. Classes can be specified also through property restrictions:$\exists R.C$ denotes the set of all objects that are related through property R to some objects belonging to class C at least one; if we want to specify to how many other objects an object is related we should write: $\leq nR$, $\geq nR$, $= nR$ where n is any natural number.

SPARQL Protocol and RDF Query Language is the RDF query language that allows retrieving and manipulating data conforming to the RDF data model. It is described by a set of specifications from the W3C and it considered a fundamental part of the semantic Web. It allows describing the condition of a query using triple patterns which are similar to RDF triples but may contain variables to add flexibility to the matching mechanism.

Semantic and data models for the IoT are required to offer more sophisticated sensor descriptions, definition of spatial and temporal concepts, relations among the different resources, etc.

The ontology-driven approaches are growing because of the power behind the semantic representation of linked data, including the description of resources and devices.

SSN-XG The W3C Semantic Sensor Network Incubator Group designed an OWL ontology called SSN-XG that describes sensors in a domain-independent way and as such facilitates semantic interoperability between sensors in sensor networks on the IoT (Compton et al. (2012)).

This classification can be used in the IoT to describe sensors and make their semantic representation globally available. One main focus of the SSN project is to develop ontologies for describing sensors and sensor networks. The second focus is the semantic annotation of sensor descriptions already available. Therefore, the SSN-XG realizes an extension of the Sensor Markup Language (SML), which is one of the four Sensor Web Enablement (SWE) languages defined by the Open Geospatial Consortium (OGC). This way, a support for semantic annotations of sensors described according to the OGC standard is realized and the combination of different services and applications becomes possible.

As reported by (Pirrò et al. (2012)), the SSN ontology offers a sensor perspective with a focus on just what senses, how it senses and what it is sensing, a data view with a focus on observations and metadata, a system view with a focus on systems of sensors and a feature view with a focus on physical features, properties of them, what can sense them and what observations of them are made. Sensors in SSN-XG are described as entities that follow sensing methods and have a feature of interest. Sensor entities may be physical devices but can also be processes and methods that observe some certain phenomena. Because of the event-based nature of sensors and sensor networks, SSN-XG further considers temporal relationships. For grouping sensors, the SSN-XG ontology provides the "system" concept. A system can further be composed of sensors or broken down into several subsystems. The process module of the ontology further opens the door to defining the function that is implemented by the described sensor. Other main concepts of the ontology describe the measurement capabilities of modeled sensors as well as the situations that are observed, i.e. the observations and the associated observation data.

2.6 Message Queue Telemetry Transport (MQTT)

MQTT is an open message protocol for M2M that enables the transferring of telemetry-style data in the form of messages from pervasive devices, along high latency or constrained networks, to a server or small message broker. Pervasive devices may range from sensors and actuators, to mobile phones, embedded systems on vehicles, or laptops and full scale computers.

There are a couple of specifications for the MQTT protocol. MQTT v3.1 specification enables a publish/subscribe messaging model in an extremely lightweight way. It is useful for connections with remote locations where a small code footprint is required and/or network bandwidth is at a premium. Based on the MQTT v3.1 specification there is an OASIS standardization process which was started in March 2013 to make MQTT an open, simple and lightweight standard protocol for M2M telemetry data communication.

MQTT-S v1.2 specification for Sensors is aimed at embedded devices on non-TCP/IP networks, such as ZigBee. MQTT-S is a publish/subscribe messaging protocol for Wireless Sensor Networks (WSN), with the aim of extending the MQTT protocol beyond the reach of TCP/IP infrastructures for sensor and actuator solutions.

MQTT is not fully overlapping with CoAP and Lightweight OMA DM, since MQTT is a telemetry protocol. CoAP is a resource access protocol, and LWM2M is a device management protocol. Therefore, they are not designed to satisfy the same requirements and use cases.

MQTT is more focused on publishing events with a subscribe mechanism. CoAP is designed to integrate RESTful architectures in constrained environments, i.e., a constrained HTTP version.

MQTT is focused on publishing events and data to a broker, while CoAP is more focused on the IPv6 principles of the IoT to provide end to end communication with the IoT resource, which is not possible with a centralized solution such as MQTT.

The centralized approach allows a full pull-based protocol for client to server, and server to client communications. Thereby,

the delay and network overload from poll approach is reduced. HTTP is not offering a push approach from the server to the client, but since it has been considered a major requirement for the IoT, CoAP is offering a push approach from the sensor/server to the client through the conditional observe solution presented in (Jara et al. (2012b)).

In conclusion, due to the centralized architecture of MQTT, even when considering that it will play a key role in the telemetry market, it will not reach a critical mass in the rest of the use cases and application scenarios where the IoT and M2M are involved.

2.7 Open Building Information Exchange (oBIX)

oBIX is a OASIS standard and aims providing the groundwork for a M2M Web using enterprise friendly technologies such as XML, HTTP and URIs. The whole design is aligned to RESTful client-server interaction using Web technologies. It further provides a standardized XML syntax for representing M2M information and a normalized representation of typical features found in automation systems like data point semantic, histories and alarming. The oBIX 1.1 working draft also includes a custom oBIX binary protocol binding in order to use oBIX on constrained devices and within constrained networks like 6LoWPANs. The oBIX TC is also currently working on a binding of oBIX on CoAP and on a encoding using EXI.

An object model provides a standard meta-model for representing device information. It uses an object-oriented approach, where everything being a device object, data point object or history or alarm is represented as an oBIX object. There are basic *value objects* like `bool`, `int`, `real` used to store a simple value, typically mapped to an I/O signal of a sensor/actuator or a soft data point like a temperature set point.

Furthermore, standard services to observe objects, query histories and alarming are specified using this contract mechanism. The contract mechanism also allows to define a standard representation for device types. However, such contracts are not part of the core specification and it is up to the vendor of an oBIX server to define its own contracts.

3. EVALUATION

3.1 Comparison of standard capabilities

This section provides a qualitative comparison of different application layer IoT standards based on a literature research. The used criteria are explained in the following paragraphs and the comparison is provided in a tabular form.

The *information modelling* criterion refers to the capabilities of the used meta modelling approach to express information. This includes *comprehensiveness*, *flexibility*, *extensibility*, *semantic capabilities* and *complexity* which refer to the fact of how many concepts are provided by the meta model. For example is an object oriented modelling approach available and whether only generic concepts are provided or the meta model is already aligned to certain domains. Furthermore, how easy can the meta-model be modified or extended and which semantics can be provided for human beings and direct machine processing and finally how complex is it to use the technology in practice.

For the provided *communication services* the amount of services (data access, device management and configuration, discovery,

		IPSO App. Frame.	MQTT(-S)	oBIX	oneM2M	ETSI M2M	ETSI ISG CIM
Inf. modeling	Comprehensiveness	Low	Medium	Medium			
	Flexibility	Low	Medium	High	High	High	High
	Extensibility	Low	Medium	High	Medium	Medium	High
	Semantic capabilities	Low	Medium	Medium	High	High	High
	Simplicity	High	Medium	Medium	Medium	Low	High
Comm. services	Amount	Low	Medium	Low	High	High	High
	Transport	HTTP, CoAP	TCP, UDP	HTTP, CoAP	HTTP, CoAP	HTTP, CoAP	HTTP, CoAP
	Encoding	Plaintext, JSON	MQTT Binary	XML, oBIX Binary [EXI, JSON]	[XML, JSON]	[XML, JSON]	[Plain Text, JSON]
	Security	None (Transport)	Weak	None (Transport)	High (Application)	High (Application)	High (End-to-End)
Maturity	Avail. implementations	Low	Medium	Medium	Medium	Avail. implementations	Avail. implementations
	Industry adoption	Medium	Medium	Low	High	High	High
	Standardization status	Not standardized	Upcoming OASIS standard	OASIS standard	OMA, 3GPP and ETSI standards	ETSI, OMA and Broadband Forum standards	ETSI

Table 1. IoT standard comparison

Criterion	Plain Text	JSON	XML	RDF	EXI	EXI(schema-informed)	Custom binary
Information Encoding	Text	Text	Text	Text	Binary	Binary	Binary
Encoding Efficiency	Medium	Medium	Low	Low	High	Highest	Highest
Communication partner coupling	No	No	No	No	No	Yes	Yes
Standardized	No	No	Yes	Yes	Yes	Yes	Yes
Platform independency	High	High	High	Medium	Medium	Medium	Low

Table 2. Message encoding evaluation

...) are accounted and the possible transport mechanisms and encodings are outlined. The built-in security capabilities are also used as a criterion.

Finally, the maturity is evaluated by comparing the amount of *available implementations*, *industry adoptions* and *standardization status*.

3.2 Comparative of the resource representation and data formats

The physical and sensor layers things are often device specific or proprietary regardless of the data format. But for the SWoT, it is crucial to support similar resource representations and data forms. In general the *information encoding* could either be text-based or binary-based. Whereas text-based encodings are desirable for human interaction and allow for investigating exchanged messages with standard tools, binary encodings are far more efficient for machine-to-machine communication. The *encoding efficiency* reflects the ratio between the pure information payload and the overhead introduced with the encoding. For example some encodings (e.g. XML-based) are rather verbose, since meta-information might be provided in a redundant way within a message. If meta information related to exchanged messages is separately exchanged in order to keep the message format small, a strong *communication partner coupling* is introduced, since the message formats have to be kept synchronized between all communicating entities. Encodings should be *standardized* in order to provide long-term interoperability and should provide *platform independence* by not being limited to a specific platform.

4. DISCUSSION

4.1 IoT evolution from heterogeneity to harmonization

The main challenge that arise for the IoT is to make a proper usage and exploitation of the IoT potential to build more powerful applications and services.

The support for heterogeneous and legacy devices integration is being integrated thanks to the IoT context brokers and middlewares that enable the interfacing of heterogeneous protocols through a homogenized and harmonized interface. Additionally, these entities such as context brokers are enabling the capacity to integrate more details about context that facilitates the exploitation of the data and content provided by the sensors with knowledge engineering technologies.

For that reason, the current steps for the IoT are focused on the importance of metadata to build intelligent solutions; and there is where emerging context-aware systems such as the proposed by ETSI ISG CIM will play a key role.

4.2 Vertical solutions to open market

The market is moving from vertical solutions where the sensors are stove-piped (one device per application) to specific platforms for its application in pre-defined use cases towards a more open market, where the sensors will be re-used, shared and accessed by a wide range of different applications.

The pending challenges cover the development of tools and protocols for dynamic interoperability, semantic discovery reasoners, mechanisms to re-adapt devices in case of change of context,

ontologies repository, and in general toolkits that allows the semantic integration and exploitation from the IoT.

Thereby, the semantics will be managed through the different phases inside of a use case, their heterogeneous devices integrated, and they will be defined interfaces among the different components involved.

Finally, the cloud is seen as the rendezvous point among different applications to collect and provide data. Thereby, a more scalable and technology agnostic solution is defined.

4.3 Proprietary solutions to open protocols

The market is also moving from proprietary solutions and protocols to open approaches such as HTTP, OMA LwM2M, CoAP (IPSO and oneM2M), and MQTT (Eclipse Foundation).

RESTFul architecture supports a resource-oriented solution, simplifying and optimizing resource manipulations for a broad range of devices and solutions, that enabled a quick and efficient application development.

Consolidated protocols such as MQTT, the emerging medium network such as LoRAWAN etc. are presenting a clear dominance of the Open Protocols with respect to proprietary options. For that reason, the support by standardization bodies that promote openness of specifications and Open Source implementations are the leaders and trackers for IoT market. Some examples are Orion Context Broker from FIWARE, MQTT, oneM2M and OMA LwM2M from Eclipse Foundation and IoTivity from the Linux Foundation.

5. CONCLUSIONS

The current status and evolution of the IoT is driven by the semantic and context-aware data models. Semantic Web of Things and the described and analysed standards in this work are presenting how to reach the interoperability and collaboration directly among them. The data exploitation of the data is the next step after of the provisioning of architectures and solutions with context-awareness. Interoperability and semantic-annotated models definitely increase the re-usability of the IoT resources outside the use cases and scope in which they were originally deployed and designed. This is a key need for Smart Cities use-cases and emerging IoT markets focused on large scale pilots. An example of data opportunities is Synchronicity (`http://synchronicity-iot.eu/`), a data marketplace for Smart Cities.

The future is unpredictable but the power of the data provided by all the resources that are being connected to Internet will bring a new conception of the world, where the semantic is required to describe what everything is, provides, and needs. To sum up, the evolution of the IoT is described by the path of six Cs, i.e., connectivity, content, context, collaboration, cloud and cognition.

ACKNOWLEDGMENTS

References

Berners-Lee, T., Fielding, R. and Masinter, L., 2005. Rfc3986 - uniform resource identifier (uri): Generic syntax. In: *Network Working Group - Internet Engineering Task Force (IETF)*.

Chu, N., Raouf, D., Corlay, B., Ammari, M., Gligoric, N., Krco, S., Ognjanovic, N. and Obradovic, A., 2013. Oma dm v1. x compliant lightweight device management for constrained m2m devices. `http://openmobilealliance.org/about-oma/work-program/m2m-enablers/`.

Compton, M., Barnaghi, P., Bermudez, L., García-Castro, R., Corcho, O., Cox, S., Graybeal, J., Hauswirth, M., Henson, C., Herzog, A. et al., 2012. The ssn ontology of the w3c semantic sensor network incubator group. *Web Semantics: Science, Services and Agents on the World Wide Web*.

D. Fernandez, G. M. H. U., 2017. Smart pois as a practical example of fiware data models and etsi isg cim. `https://www.smartsdk.eu/2017/02/16/smartpoi/`.

Davis, E. G., Calveras, A. and Demirkol, I., 2013. Improving packet delivery performance of publish/subscribe protocols in wireless sensor networks. *Sensors* 13(1), pp. pp. 648–680.

ETSI, T., n.d. Etsi ts 102 921 v1.1.1 (2012-02) - machine-to-machine communications (m2m); mia, dia and mid interfaces. *Machine-to-Machine Communications (M2M)*.

FI-WARE, 2012. Etsi m2m mid open restful api specification.

Horrocks, I., Patel-Schneider, P. and van Harmelen, F., 2003. From SHIQ and RDF to OWL: the making of a Web Ontology Language. *Web Semantics: Science, Services and Agents on the World Wide Web* 1(1), pp. 7 – 26.

Hui, J. and Thubert, P., 2011. Rfc 6282: Compression format for ipv6 datagrams over ieee 802.15.4-based networks. *IETF*.

Interdigital, 2012. White paper: Standardized machine-to-machine (m2m) software development platform.

Jara, A. J., Moreno, P., Skarmeta, A., Varakliotisy, S. and Kirstein, P., 2013. Ipv6 addressing proxy: Mapping native addressing from legacy technologies and devices to the internet of things (ipv6). *Sensors*.

Jara, A. J., Zamora, M. A. and Skarmeta, A., 2012a. Glowbal ip: An adaptive and transparent ipv6 integration in the internet of things. *Mobile Information Systems* 8(3), pp. pp. 177–197.

Jara, A., Li, S. and Hoebeke, J., 2012b. Conditional observe in coap. *IETF*. `http://tools.ietf.org/html/draft-li-core-conditional-observe-03`.

Pirrò, G., Talia, D. and Trunfio, P., 2012. A dht-based semantic overlay network for service discovery. *Future Generation Computer Systems* 28(4), pp. 689–707.

Ruta, M., Di Noia, T., Di Sciascio, E. and Scioscia, F., 2007. If objects could talk: A novel resource discovery approach for pervasive environments. *International Journal of Internet Protocol Technology* 2(3), pp. 199–217.

Shelby, Z. and Chauvenet, C., 2012. The ipso application framework. draft draft-ipso-app-framework-04. `http://www.ipso-alliance.org/wp-content/media/draft-ipso-app-framework-04.pdf`.

Shelby, Z., Hartke, K. and Bormann, C., 2013. Constrained application protocol (coap). `http://tools.ietf.org/html/draft-ietf-core-coap-14`.

SmartSDK, 2011. A fiware-based sdk for developing smart applications. `http://www.gsc16.ca/english/documents/openplenary/GSC16-PLEN-42a1r1.xlsx`.