




Semantic Middleware Architectures for IoT Healthcare Applications

Rita Zgheib¹ , Emmanuel Conchon² , and Rémi Bastide³ 

¹ Université de Pau & Pays Adour, E2S-UPPA, LIUPPA, 64600 Anglet, France
rzgheib@iutbayonne.univ-pau.fr

² Université de Limoges, XLIM UMR CNRS 7252, 87060 Limoges Cedex, France
emmanuel.conchon@xlim.fr

³ Université de Toulouse, ISIS-IRIT, Campus Universitaire, 81104 Castres, France
remi.bastide@irit.fr

Abstract. The adoption of the Internet of Things (IoT) in healthcare has received considerable interest in the past decade. Indeed, IoT-based solutions are poised to transform how we keep people safe and healthy especially as the demand for solutions to lower healthcare costs increases in the coming years. However, the heterogeneity of the *things* that can be connected in such environments makes interoperability among them a challenging problem. Moreover, the observations produced by these things are made available with various vocabularies and data formats. This heterogeneity prevents generic solutions from being adopted on a global scale and makes difficult to share and reuse data for other purposes than those for which they were initially set up. In this book chapter, we provide an overview of the different solutions from both technical and semantic perspectives that have been used recently to tackle the interoperability issue in such IoT environments and especially in healthcare domain. We also present an overview of semantic middleware solutions that have combined the technical and semantic techniques for a complete interoperable solution.

Keywords: Semantic architecture · Ontology · Middleware
Internet of Things

1 Introduction

With the evolution of smart connected devices in the last decades, the world keeps asking “How smart will the Internet of Things be?”. We all know that sensors are already embedded in all sorts of objects, machines, and things and that many of those sensors are communicating with other machines over the Internet. IoT is a reality today, and its power on improving the quality of life and business is quite remarkable. In this context, building IoT-based healthcare applications provides the possibility to improve people lives. However, IoT in

This chapter is an extended version of Chap. 2 of Rita Zgheib’s Ph.D.

© The Author(s) 2019

I. Ganchev et al. (Eds.): Enhanced Living Environments, LNCS 11369, pp. 263–294, 2019.

https://doi.org/10.1007/978-3-030-10752-9_11

healthcare domain presents several challenges related to the fact that almost every week a significant vendor announces a new IoT strategy or division. The proliferation of ad-hoc and specific healthcare products, sensors and applications pose significant challenges about the complexity of designing and managing such systems, to the heterogeneity of the generated data, to the scalability, and the flexibility of the system to support the integration of highly distributed and heterogeneous data and knowledge sources.

In the research area, several Middleware architectures have been proposed in order to manage IoT system components and mediate between sensors and IoT applications. From the other side and beyond traditional management of sensors, some projects have addressed interoperability and scalability challenges in IoT from a semantic perspective. In this trend, ontologies [101] for describing sensors data and metadata as well as describing domain description, have been widely used and found as a reliable technique due to its significant efficiency in ensuring interoperability and clarification of knowledge structure. Middleware and ontologies have mostly been proposed individually. However, IoT environments are evolving rapidly, and the needs for a complete semantic middleware architecture is becoming an essential requirement in IoT environments. Especially in healthcare where the technical with the semantic interoperability promote the reduction of the number of sensors in the person's environment and the development of smarter applications by exchanging analyzed and processed data between all components.

Recently, some projects have tried to combine middleware solutions with semantic techniques in order to have a complete interoperable IoT architecture. Few developers and researchers rely on this combination due to the complexity of designing and integrating their concepts. Our objective is to promote research in combining semantic and middleware solutions while most researches focus on advancing one topic. For that aim, we provide in this chapter an overview of several middleware approaches in Sect. 2. Then, we outline in Sect. 3 the ontologies used to describe sensors and the domain knowledge in IoT. In Sect. 4, we study projects that have combined both approaches in their proposals, we show their characteristics and their advantage regarding traditional architectures for IoT management. Finally, we present the ongoing challenges in providing the most relevant semantic middleware for IoT healthcare applications.

2 Middleware Architectures for IoT Systems

Middleware [11] solutions provide a technical infrastructure that mediates between two or more systems. Their historical role is to ensure the transport of a message from one subsystem to another with a more or less important level of coupling. Researchers find middleware as a suitable solution to fill the gap of heterogeneity and manageability of sensors because it provides an abstract layer positioned between the network layer and the application layer as it is shown in Fig. 1. Moreover, It aims to hide the technological details of communication technologies to enable the application developers to focus on the development of the IoT applications.

2.1 Middleware Challenges

Several middleware architectures have been proposed in the literature trying to address the infrastructure challenges and the applications challenges in IoT. They are regarded as essential problems to be solved before presenting an approach as a final solution.

The infrastructure challenges refer to (1) the **interoperability** issue as heterogeneous devices will communicate and exchange information together; (2) the **scalability** issue since a large number of devices are expected to be supported by IoT application; (3) the **spontaneous interactions** of objects and devices; (4) the **diversity of infrastructure** since IoT devices have a specific infrastructure (mobile, wirelessly connected, etc.), and resource constrained which gives the IoT network a **dynamic behavior**; and (5) the need for **abstraction** at many levels such as the physical layer, interfaces, data stream, and development process.

The application challenges consist of guaranteeing (1) the **availability** of services and information at all time; (2) a high level of system **reliability** that should remain operational even in the presence of failures; (3) a **real-time** information delivery especially for critical domain such as healthcare; (4) and finally the **security and privacy** as much information is shared between IoT components. This information can be private and even personal such as with information about daily life.

The common goal of all these middleware development initiatives is to develop a framework which can enable an adaptation layer in a plug-n-play mode. Despite that, each middleware architecture [94] focuses on some of the challenges cited before and takes into account some application's requirements which differ from each other's regards to IoT domains they target. In light of the presented challenges, we provide an overview of these approaches emphasizing their characteristics.

2.2 Overview of Middleware Approaches for IoT

Several studies on middleware approaches for IoT have been proposed like the classifications presented in [92,93]. A specific study is presented by Razzaque et al. in [94]. In this paper, authors have analyzed in depth several existing solutions and have presented a recent overview of the different kinds of middleware architecture. Thanks to this study, we have identified the different types of middleware (see Fig. 1) for intelligent environments that have been classified as follows: Application-specific, Agent-based, VM-based, tuple-spaces, database-oriented, Service-oriented architecture and Message-oriented Middleware.

Application-Specific: This approach is based on the needs of a specific application and focuses on resource management. Thus, a strong coupling exists between application and data providers which leads to specialized middleware.

MidFusion [8] represents an example of this middleware approach. It discovers and selects the best set of sensors or sensor agents on behalf of applications.

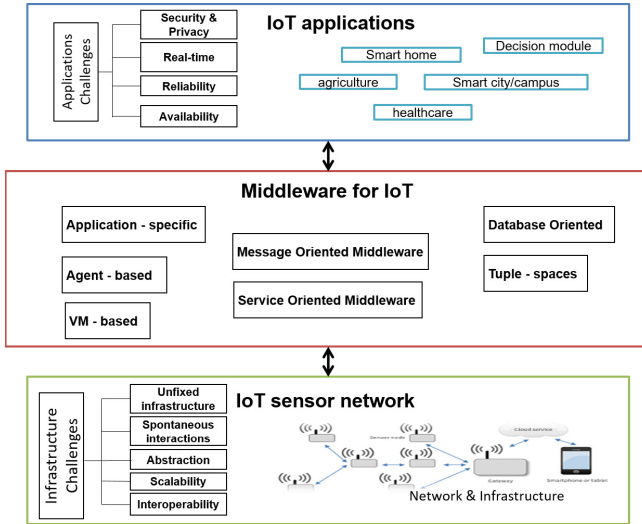


Fig. 1. The Relationship between IoT Data sources and IoT applications via the middleware mediator

It provides a sensor selection algorithm to select the best set of sensors using the principles of Bayesian and Decision theories. Its limitation is that it provides a (Quality of Service (QoS) support only for networks based on Bayesian algorithms. Other middleware in the same category can be found in [60] which focuses on applications in smart homes. It uses application-specific functions to choose a specific context. Given multiple alternatives, one alternative at any time provides the context for all applications, whilst maximizing the applications total satisfaction with the quality of context from the chosen provider. In [55], authors propose MiLAN, a middleware based on tight coupling between the components. It allows applications to specify a policy for managing network and sensors. They argued that the needs of the application should be integrated with the management of the network into a single unified middleware system.

This kind of architecture does not satisfy all IoT middleware requirements due to its tight coupling nature between applications and data providers. It does not address the heterogeneity characteristic of the IoT environment.

Agent-Based: Agent or modular based approach [73] consists in applications division into modular programs to promote distribution and injection through the network using mobile agents. Impala [71], Smarty messages [66] and Agilla [46] are examples of this approach. They can be highlighted by providing decentralized systems capable of tackling the availability, reliability, and resource management requirements of middleware. This approach can reduce the complexity of a middleware architecture. However, it presents some limitations related to its inability to perform code management tasks and the unpredictability of agents in the system at runtime.

VM-Based: This approach is flexible and contains virtual machines (VMs), interpreters, and mobile agents. The middleware is then composed of two layers. Each physical device is deployed as a VM in the first layer of the middleware. In the second layer, a general VM interprets the modules and delivers data to the application that expresses its needs with a query. This approach addresses architectural requirements such as high-level programming abstractions, self-management, and adaptivity while supporting transparency in distributed heterogeneous IoT infrastructures [58, 74]. However, this approach suffers from the overhead that the exchanged instructions introduce.

Tuple-Spaces: A tuple space [82] is a data repository that can be accessed concurrently. Each device from the physical layer is represented as a tuple-space in the middleware. All tuple-spaces form a federated tuple space on the gateway. This approach suits mobile devices in an IoT infrastructure as they can transiently share data within gateway connectivity constraints. TinyLime [34] and TeenyLIME [33] are tuple-space middleware solutions for mobile ad hoc networks and sensor networks. Although they have a flexible architecture that allows middleware to be used in different environments, they address frequent disconnections and asynchronous communications problems. However, they offer limitations in resource management, scalability, security, and privacy.

Database-Oriented: This middleware approach considers the whole sensor network as a distributed and virtual database. It uses SQL like queries to collect data over the network. GSN [4] is a database-oriented middleware that has been integrated into other projects such as OpenIoT [68]. While this approach offers good programming abstraction and proper data management support, the remaining IoT requirements are not necessarily addressed like scalability, real-time and spontaneous interactions. Moreover, its centralized nature makes it difficult to handle dynamic and heterogeneity characteristics of the IoT network.

The architectures as mentioned above have been used in many specific IoT projects and research area. However, the important use of Service Oriented Architecture (SOA) and Message Oriented Middleware (MoM) solutions is remarkable in IoT projects in the past decade.

Service-Oriented Architecture (SOA): Two major trends in the world of IoT have been witnessed in the past years [49]. First, the hardware is becoming smaller, cheaper and more powerful. Second, the software industry is moving towards service-oriented integration technologies. Service-Oriented Architecture (SOA) is a way of thinking and designing the Information System and has been traditionally used in corporate IT systems. The key concept of SOA is the service which is a distributed software invocable that can that can be quickly modified or redeployed in new contexts, allowing applications to respond to changing consumer needs quickly. In IoT approaches based on SOA, intelligent sensors are depicted as services for consumer applications. The key point with these services is their encapsulated nature (i.e., the service interface is independent of the implementation). Service providers describe their services

(sensors characteristics) and expose them to consumers. Web Services Description Language (WSDL) is the standard used for such a description [38].

A state of the art of SOA-based middleware solutions for wireless sensor network can be found in [81]. It refers to *SStreaMWare*, *USEME*, *SensorWeb 2.0*, *OASiS*, *B-VIS*, *MiSense*, *SOMDM (SI)*, *SOA-MM* as middleware proposals for wireless sensor networks. Their main features are supporting real-time monitoring, management of heterogeneous devices, data collection and filtering. However, the survey demonstrates that none of these solutions covers all the requirements of the management of sensors network in intelligent environments.

More recent approaches have tried to enhance the SOA features and adapted it to IoT. For example, SenseWrap [45] provides a standardized communication interface to hide the sensor-specific details from applications. It introduces the concept of the virtual sensor to offer transparent discovery of sensors. However, virtualization is applied only to sensors and not to actuators or computing resources which makes it not fully suitable for IoT environments. TinySOA [10] uses simple and deterministic mechanisms for WSN resource (e.g., sensor nodes) registration and discovery. It supports only a few basic functional requirements (e.g., abstraction, resource discovery, and management). SensorsMW [9] has been found as an adaptable and flexible middleware for sensors management. It allows easy and efficient configuration of wireless sensor networks for information gathering. However, the reconfiguration may fail in critical applications as they define strict QoS rules. MOSDEN [87] supports sensing as a service model built on top of GSN. It is based on a plugin architecture which improves the scalability and user-friendliness. However, the predefined resource/service discovery and service composition mechanisms features presented in this approach may be challenging in a dynamic IoT environment.

Message Oriented Middleware (MoM): A Message Oriented Middleware has been used for a long time in network communications especially in industrial networks such as integrated manufacturing systems [13,36]. It offers an event-based architecture and a publish/subscribe communication model. In event-based architecture, components, applications, and all other participants interact through events. Events are propagated from the sending application components to the receiving application components following the publish/subscribe paradigm. The publish/subscribe model is an interaction model that consists both of publishers and subscribers. Data sources (publishers) and destinations (subscribers) are decoupled from each other and data objects (messages) are filtered and delivered to destinations based on predefined topics expressed as subscriptions thanks to a dedicated component called the message broker as depicted in Fig. 2. The broker can be seen as a mediator responsible for the management of the distribution of messages to serve the right information to the right consumer. The strength of this middleware mainly lies in its support for asynchronous communication allowing a loose coupling between the sender and the receiver.

Since the apparition of the Internet of Things, the publish/subscribe mechanism has been put into the light for its effectiveness in offering loose coupling.

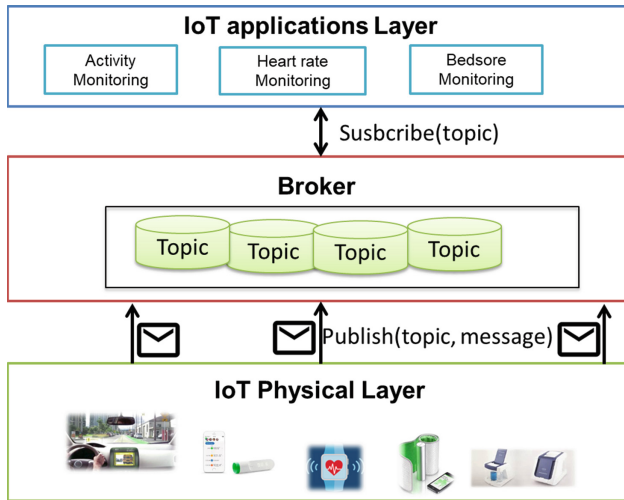


Fig. 2. General design model for an event-based middleware.

Compared to the SOA architecture which is also widely proposed for IoT solutions, a MOM follows a message-based distribution model focusing on the information. It differs from the classical client/server paradigm in that neither the source nor the destination of the message has to be known from each other before communication.

Few IoT projects have proposed publish/subscribe solutions in the literature. For instance, CenceMe project [78] aims to automatically infer people's activity (e.g. dancing in the party) based on a sensor-enabled smartphone in order to share this activity through social media like Facebook. Another example supporting easy access to sensor data on mobile phones is Pogo [21], a publish/subscribe middleware infrastructure for mobile phone sensing. It uses simple topic-based subscriptions to manage access to sensor data and reports significant energy gains due to topic-based filtering of sensed data on mobile devices.

On the other hand, a number of Cloud-based services dedicated to storing sensor-based data are nowadays available. Few examples that could be mentioned are Xively [3], ThingSpeak [2] and iDigi [1] which support connections using MQTT (Message Queuing Telemetry Transport) protocol. They represent a scalable infrastructure that enables users to build IoT products and services and to store, share and discover real-time sensor. To sum up, a comparative table is presented in Table 1. In the following section, we outline middleware solutions developed in the healthcare context.

2.3 Middleware Solutions in Healthcare

Healthcare is an active research topic in IoT where applications can be classified into Medical applications and Ambient Assisted Living (AAL) applications.

Table 1. Comparative table of Middleware approaches

IoT challenges		Middleware approaches						
		Application-specific	Agent-based	VM-based	Tuples-spaces	Database-oriented	SOA	MoM
Infrastructure challenges	Technical interoperability		X	X			X	X
	Scalability							X
	Spontaneous interactions			X	X			X
	Unfixed infrastructure		X	X	X		X	X
	Abstraction	X	X	X	X	X	X	X
Application challenges	Availability	X	X	X	X	X	X	X
	Reliability		X	X		X	X	X
	Real-time						X	X
	Privacy & security							

Medical applications are mainly installed at hospitals and retirement homes to prevent, detect diseases or even monitor patients health condition. AAL applications are mainly installed at a patient's home to monitor and follow his/her health and activity at home. In this context, middleware solutions have been proposed to fulfill the requirement of IoT and healthcare domains.

- **Middleware for Medical applications:** In this strand we find the *Sphere* project [120] that follows a clustered-sensor approach. It aims to build a generic platform that fuses corresponding sensor data to generate rich datasets that support the detection and management of various health conditions. LinkSmart [85] within the REACTION European project is another SOA-based middleware with the aim of monitoring and managing the diabetes of patients as well as their therapy in operational healthcare environments. It has also been used recently for a smart home application [107]. A middleware for bedsores detection can be found in [86] where authors defined a SOA architecture for bedsores detection and sleep monitoring. The main contribution of this work is about collecting information from wireless and wearable sensors. Other monitoring system proposal can be found in [77] where body sensors are connected to a smartphone via Bluetooth to get information like heart rate and body temperature. MyHealthAssistant [103] is an Event-driven Middleware for Multiple Medical Applications. Its objective is to merge information from several wireless sensors on a Smartphone-mediated Body Sensor Network.

Uranus [31] is a SOA-based middleware architecture for dependable AAL and vital signs monitoring applications. It provides a rapid-prototype for monitoring the level of oxygen in the blood of a chronically ill patient. It also describes another prototype for monitoring patients in the context of a smart hospital. MyHealthAssistant [103] presents an event-driven middleware targeted for medical applications on a smartphone to enable flexible coupling

with changing sets of wireless sensor units. Waluyo et al. [114] propose a middleware for medical Body sensor network that supports multiple sensors and applications, plug and play features, and resource management. In that project, however, parts of the middleware and the applications are hosted on a single PC.

- **Middleware for AAL applications:** SM4ALL [115] has been developed and build on top of the OSGi/UPnP standards for building smart homes in the context of European Framework 7 project and Smart Homes for All (SM4ALL) project. It targets Ambient Assisted Living (AAL) environments to monitor disabled elders. VIRTUS [12] is an event-driven middleware based on the standard XMPP (eXtensible Messaging and Presence Protocol). It guarantees a (near) real-time, secure and reliable communication channel among heterogeneous devices. In [43, 54], the authors present a platform based on cloud computing to manage mobile and wearable healthcare sensors that demonstrates that the IoT paradigm can be applied to pervasive healthcare. In [30], a SOA based solution offering a distributed telemonitoring system that aims at improving healthcare and assistance for dependent people in their homes. In [110], authors present a pervasive health system integrating patient monitoring, status logging, and social sharing enabling self-management of chronic patients in their environment. Junnila et al. propose in [64] in-Home health monitoring platform with a *common sensor interface* architecture that supports a large number of Zigbee sensors. The platform is tested with two health related case studies: one with an senior woman living in sheltered housing and the other with a hip-surgery patient during his rehabilitation phase.

2.4 Discussion

Middleware proposals based on Application-specific, Agent-based, VM-based, tuple-spaces and database-oriented, have tried to tackle the requirements of manageability and heterogeneity of IoT environments. These approaches have been used in specific IoT applications, and their limitations stated in each case make them not so popular in this environment. Moreover, it is clear that the most used approach in the IoT world is SOA-based architecture, in particular, most healthcare solutions rely on SOA-based middleware. However, we recognized that even if SOA presents a good solution for IoT healthcare applications, it can not meet all IoT requirements especially related to heterogeneity, scalability, real-time processing and flexibility. From the other side, we recognized that although a few recent studies have considered the MoM approach, it has been recognized as a flexible and suitable communication model in IoT. Based on the before mentioned middleware solutions, a comparison between SOA and MoM shows that:

- In regards to SOA architecture, MoM follows a message-based model focusing on the information itself and supports sending and receiving of messages between distributed systems. While in the SOA approach, data providers

and data consumers agree on a service contract before data flow sharing as it is evident in Fig. 3, this approach makes the communication not completely decoupled between system components.

- Despite its historical role in sensors management and its potential benefits in IoT middleware solutions, SOA-based solutions focusing on services provided by the system do not scale well in ultra large and dynamic IoT environments. While on the other side, publish-subscribe approaches are gaining more momentum due to their ability to promote scalable, flexible and fully decoupled communication in intelligent environments. As a result, we believe that a MoM with the publish/subscribe paradigm is the most suitable middleware solution for IoT healthcare applications.

To sum up, middleware solutions have been found to address technical interoperability issues and communication requirements in an IoT environment. However, the heterogeneity of connected devices and the variety of data sent over the network introduce the requirement of providing a unified and coherent manner of representing data among system components. Moreover, in healthcare, it is necessary to take into account the patient comfort by reducing the number of sensor around him. This can be done by improving data sharing between IoT application. Although the benefits of middleware solutions in virtualizing the technological details and infrastructure, providing a semantic description would enhance the interoperability and provide a full abstraction of the technical level. In the following section, we present how semantic technologies have been integrated into IoT.

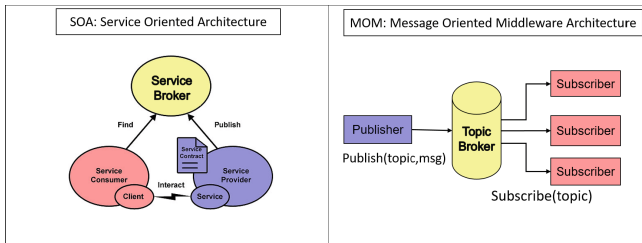


Fig. 3. Service Oriented Architecture vs Message Oriented Middleware Architecture

3 Semantic Description in IoT

Different data providers send sensor data in IoT environments and received by many data consumers. Querying sensors and information sharing present a challenge for IoT applications due to the massive number of heterogeneous sensors with specific characteristics. To address this issue, research initiatives and standardization activities have mainly focused on modeling sensors observations and on sending the semantic observations over the network. In [35], different data

models have been presented as a coherent and reliable way to share information across an IoT system: XML, Web Service Definition Language (WSDL), JavaScript Object Notation (JSON), Resource Description Framework (RDF) and Web Ontology Language (OWL). RDF and its new version, the OWL format, has become the most used technique for sensor data description due to their ability to ease both reasoning over sensor data and semantic interoperability among devices [51, 109].

The Web Ontology Language (OWL) built upon a W3C XML standard is a Semantic Web language designed to represent rich and complex knowledge about things, groups of things, and relations between things. It can easily be integrated with many programming languages like Java through APIs (Jena or OWL-API). Ontologies in IoT have an essential role in representing sensors networks (physical aspect and infrastructure) and also in describing complex domain knowledge and concepts. Furthermore, the main advantage of using ontologies is the possibility to generate OWL representations which provide the opportunity to exchange formal messages at the technical layer in OWL format. Describing data in a formal language such as OWL guarantees the reliability of information and improves the interoperability of the system. Finally, it fills the need to fuse heterogeneous sensor data and potentially inaccurate data. In this section, we present an overview of the several ontologies used to describe sensors in IoT, and we also outline ontologies that describe the healthcare domain.

3.1 Sensors Modeling and Description

Sensors description has been proposed from one hand to promote reusability and integration of sensors and from the other hand to help to solve the difficulties of installing, querying [7] and maintaining complex, heterogeneous sensor networks. In this field, XML schema has been used for sensors description such as in sensor model language (SensorML) [18] that provides metadata model to describe sensor capabilities and measurement process. They are also used by SOA-based middleware solutions to describe sensors characteristics as a service contract to data consumers. Also, ontologies [101] for sensors description have been widely used as a reliable technique due to their efficiency in ensuring both interoperability and clarification of knowledge structure.

The main advantage of ontologies is their ability to describe three perspectives of a sensor [5]. First, it describes sensors metadata, sensor type, components, configuration, process, and properties. Second, it allows to describe what a sensor can measure after a stimulus is detected or to describe the observation of the sensor in term of frequency, accuracy and measurement capabilities. Third, it provides the concepts to describe data stream and the observation result, allowing semantic reasoning, generating new knowledge and detecting inconsistency in IoT environments.

Key standardization efforts that have sought to establish sensor data models for sensors to be accessible and controllable via the Web include the OGC Sensor Web Enablement (SWE). The SWE efforts established by the Open Geospatial

Consortium include the following essential specifications: Observation & Measurement (O&M), Sensor Model Language (SensorML) and Sensor Observation Service (SOS) [19]. The O&M and SensorML contain standard model and XML schema for observations/measurements and sensors/processes respectively. The SOS is a standard service model, which provides a mechanism for querying observation and sensor metadata. Based on these standards, several ontology approaches have been developed to describe sensors. Compton et al. [29] and Bendadouche et al. [15] have surveyed the different attempts aiming at establishing a sensor ontology covering the description of all sensor topics.

In 2008, four potential ontologies have been proposed CESN, SWAMO, A3ME, and ISTAR. The Coastal Environmental Sensing Networks (CESN) [22] project for sensor networks for coastal observing has built an ontology to describe relationships between sensors and their measurements. It also provides logic programming rules reasoner to validate sensor observation and to test anomalous sensor observation by a decision maker. SWAMO [116] describes physical devices and process models and tasks in distributed and intelligent software agents environment. Also, A3ME ontology [57] was developed to classify devices and their capabilities in a heterogeneous network. The ISTAR [47] ontology was developed as part of a system to automatically select sensors for tasks based on their fitness for the task description.

In 2009, OOSTETHYS [16], MMI [100] and CSIRO [84] ontologies were proposed. The OOSTETHYS ontology has been designed to describe sensors and get observation and capabilities for oceanographic observing. Likewise, The Marine Metadata Interoperability (MMI) Device Ontologies Working Group has developed the MMI ontology of oceanographic devices, sensors and samplers. The CSIRO ontology is a generic ontology for describing sensors and deployments. It is intended to be used in data integration, searches, and classification features. It can express complex compositions and finds details of the function and results of sensors and processes. The ontology can also encode much of the information in SensorML documents.

Ontologies developed until 2009 are either specific to a domain (oceanographic, ecology, etc.) or discontinued. Therefore these efforts did not lead to a mature and general solution applied for ongoing projects. None of the ontologies can express all the properties required for a full description of sensors, and a standard is still not available until 2011.

In 2011, the W3C organism proposed SSN, the Semantic Sensor Network Ontology [28]. It has been initiated by the CSIRO, Wright State University, and the OGC as a forum for the development of an OWL ontology for sensors and to further investigate annotation of existing concepts, and links to existing standards. SSN has been designed as a generic ontology which has become one of the most popular and efficient ontology to describe sensors and observations. It has been conceived as a domain-independent ontology where extensions can be made to add domain-specific knowledge.

Extensions of SSN have been developed in the last years by adding concepts related to time, space, communication and concepts related to domain

characteristics. We review in this section a broad range of these solutions considering that the presented ontologies are just a small example of existing ontologies that are built on top of SSN. It helps to show the impact of SSN on the semantic description of sensors over the years.

BFO [61] is a spatio-temporal extension that make a distinction between describing identities that happen at a finite time and events like storm and routing. It is a hierarchical system approach where sensors collect data from the real world and send it to clusterhead-node.

In Bandadouche et al. in [15], authors present the limitation of SSN in describing the communication process of a wireless sensor network. They propose a new ontology design pattern Stimulus-WSNnode-Communication, an extension of SSN, that addresses the communication limitation by integrating new concepts that describe the communication process of a wireless sensor network. Another extension for SSN based on fuzzy logic is proposed in [14] to support fault tolerance and for large-scale Wireless Sensor Network. It is a service-oriented approach to build diagnosis and test services for wireless sensors.

Authors in [96] present another SSN extension ontology for WSN, and it is presented as an alignment of many ontologies SSN, SWRLTO, TAO, and DOLCE. This solution aims to improve time descriptions limitation in SSN by representing temporal abstraction to analyze data in real time. SSN is used for sensors measurements; SWRLTO is used for temporal modeling and reasoning; and TAO designed by the authors of this paper to capture the semantic TA (Temporal Abstractions). This framework uses temporal reasoning to search and classify temporal patterns that help to infer the processed data. For the alignment of the three ontologies, the authors use DOLCE, a known upper ontology.

Recently, in [104], authors have analyzed general IoT ontologies: SSN, Smart Appliance REference (SAREF)¹, IoT-ontology², IoT-lite³, Spitfire⁴, IoT-S⁵, SA⁶ and the oneM2M ontology⁷. They did not consider specific domains impacted by IoT (domotics, agriculture, smart cities...) but they studied ontologies that they found on the web. Based on their comparative study, none of the presented ontologies can describe actuators. Actuators are devices that transform an input signal into a physical output, making them the exact opposite of sensors. They have proposed the IoT-O ontology in order to describe actuators, services and energy consumption.

In October 2017, the last version of SSN ontology was recommended by the World Wide Web Consortium (W3C). The main innovation of this version of SSN has been the introduction of the Sensor, Observation, Sample, and Actuator (SOSA) ontology, which provides a lightweight core for SSN. SOSA aims

¹ <http://sites.google.com/site/smartappliancesproject/ontologies>.

² <http://ai-group.ds.unipi.gr/kotis/ontologies/IoT-ontology>.

³ <http://iot.ee.surrey.ac.uk/fiware/ontologies/iot-lite>.

⁴ <http://sensormeasurement.appspot.com/ont/sensor/spitfire.owl>.

⁵ <http://personal.ee.surrey.ac.uk/Personal/P.Barnaghi/ontology/OWL-IoT-S.owl>.

⁶ <http://sensormeasurement.appspot.com/ont/sensor/hachemonto.owl>.

⁷ <http://www.onem2m.org/ontology/BaseOntology/>.

at broadening the target audience and application areas that can make use of Semantic Web ontologies. With the new version of SSN, actuators and virtual sensors can be easily described which was not possible with the last version.

SSN is commonly used by many projects and is still the most appropriate ontology that can describe an IoT sensor system. Describing physical sensors with SSN promotes the reusability of sensors and makes IoT systems sustainable. Furthermore, the new SOSA ontology integrated with SSN allows having a complete description of physical and virtual sensors that can be deployed in an IoT environment. We recognize virtual sensors as any application that generates information that can not be gathered from the physical world. For example, an application that uses a motion sensor in a living room data and outputs the presence of the person can be seen as a virtual sensor in the IoT world (Fig. 4).

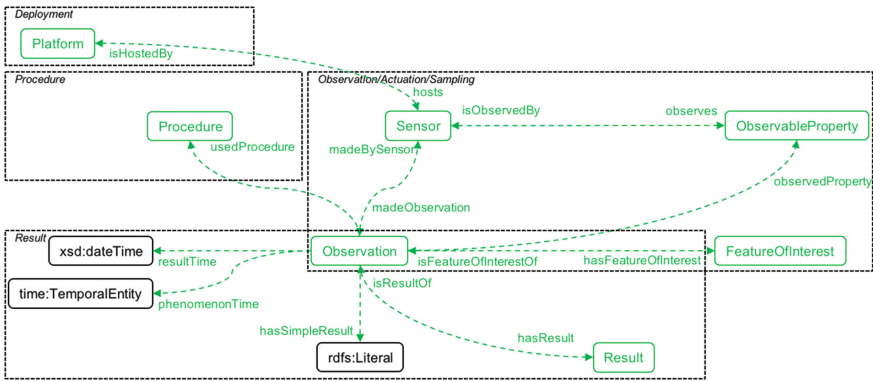


Fig. 4. Overview of the SOSA ODP

3.2 Ontologies for Healthcare Description

Domain-specific IoT applications are becoming increasingly popular and, domain knowledge description is gaining even more popularity. Domain knowledge is already defined in more than 200 ontologies and sensor-based projects [52]. These ontologies provide expert knowledge description as well as the conceptualization of the different domains like transportation or diseases. Specifically, domain applications use ontologies and semantic description as a way to define types, properties, and interrelationships of the entities that exist for a specific domain.

As an example, authors in [56] have proposed an ontology to define logistic terminologies. It is intended to assist a logistics expert in identifying and precisely specifying the logistic problem for solving a passenger train optimization problem. In the same context, in [44], an ontology is proposed for monitoring and improving public transportation. Another example of ontologies domain is related to energy concepts as presented in [102] where authors created ontologies to describe energy sources: wind power, biomass, and fossil fuels. Further

domains have relied on ontologies to describe concepts and relations that constitute their particular fields. In agriculture, we distinguish AgroPortal ontology for agriculture and nutrition data [63].

In healthcare, many ontologies have been developed to describe both medical and AAL applications.

- **Ontologies for Medical applications:** the **Disease Ontology (DO)** [67] represents human diseases for linking biomedical knowledge through disease data. Also, the well known **SNOMED-CT** [42] represents an advanced terminology and coding system for eHealth. In the same context, authors in [70] proposed an ontology to describe the **patients vital signs**. The ontology aims to create a profile for each patient containing the several medical control information. Moreover, authors in [23] present an example of **food ontology** for diabetes control. The **HL7 Security and Privacy Ontology** is another example that serves to name, define, formally describe, and interrelate key concepts within the scope of Healthcare Information Technology. Kim et al. [69] proposed an Ontology-driven Interactive Healthcare with Wearable Sensors (**OdiH_WS**) to achieve customized healthcare service. It aims to acquire context information at real-time using ontological methods by integrating external data such as a meteorological website in order to prevent disease. **ContoExam** [20] is an ontology developed to address the interoperability problem of sensor networks in the context of e-health domain applications. It contains specific expressions and specifications for medical use as examination vocabulary and expressions.
- **Ontologies for AAL applications:** In this context, Activities for Daily Living (ADL) is a contemporary topic. Researchers are working in improving the quality of life of elderly at home by providing monitoring systems. In this strand, Ontologies-based solutions have been proposed to describe human activities and behavior like in [76]. These ontologies propose a description of a set of actions and activities of daily life where the activity is inferred from several actions [26,27]. For example, going to the kitchen, taking a cup, and adding water are single actions and drinking is an activity. The CoDAMoS [37] ontology focuses on modeling roles, hardware, and software services around four main core entities: user, environment, platform, and service. A more in-depth survey of ontologies for human behavior representation can be found in [98].

Other researchers have illustrated the limitations of ontology-based solutions in describing uncertainty. For instance, Authors in [41,97] rely on fuzzy concepts to describe real-world context and uncertainty in activities where several solutions can be deduced based on the same sensors data. In the same context, an ontology for gym addict and lifestyle profiling is presented in [40].

From the other side, there is a set of ontologies that try to tack the concurrent activities. For example, the Knowledge-driven approach for Concurrent Activity Recognition (KCAR) [118] consists on segmentation of sensors data into fragments, each of which corresponds to one ongoing activity.

With respect to other solutions, knowledge-based methods are semantically clear in modeling and representation and highly effective in inference and reasoning. They create a completely accurate model of ADL [95] that enhances the semantic interoperability of an IoT system without a prior learning phase. However, there are still some issues and challenges to tackle in this field such as defining an ontology that take into account the description of actions, activities with uncertainty, concurrent and overlapping activities that have been identified in [59].

3.3 Discussion

In this section, we have discussed the several perspectives of using the semantic description in IoT. We have found that ontologies are the most common semantic technology that describes the three main concepts: sensors, data, and domain.

A well designed ontology enhances IoT environments and architectures by ensuring some features like abstraction, scalability, integration, smart reasoning and interoperability as summarized in Fig. 5. Moreover, an IoT ontology model relies on several design principles: it should be modular to facilitate its evolution, extension and integration with external ontologies; and it should be lightweight to be widely adopted and reusable. Furthermore, an ontology should be a complete reflection of a full description of an IoT environment which reduces the need to import other ontologies. The compatibility principle is related to the needs to be consistent with existing ontologies.

We have shown that SSN is a suitable ontology for sensors description in IoT and is able to generate data in RDF/OWL format. Presenting data in a formal way using RDF/OWL enhances the interoperability and homogeneity of information between IoT systems. In the context of healthcare application, several ontologies have been proposed to describe diseases and activities of daily living. We also found that requirements and challenges are still ongoing in this field as it is an active and recent topic. Moreover, the alignment of sensors ontologies and domain ontologies could be a research aspect of improving.

As we discussed before, integrating middleware approaches with semantic technologies creates a suitable platform and environment for IoT applications. It provides manageability and technical and semantic interoperability in the overall system. For that, it was cumbersome to state the several projects that used these two paradigms.

4 Semantic Middleware Approaches

4.1 Overview on Existing Semantic Middleware Architectures

Existing approaches have considered semantic description in proposing a middleware architecture for IoT applications. SOA and MoM approaches are prominent IoT architectures where semantic enhancements have been added.

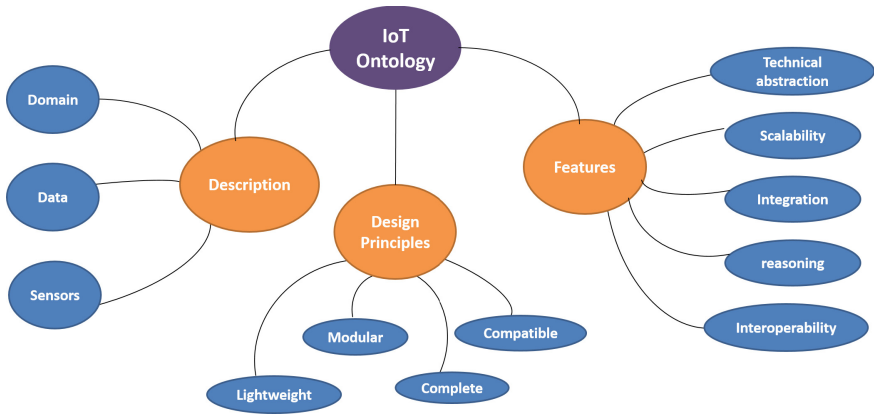


Fig. 5. Main characteristics of an IoT Ontology

Semantic SOA-Based Architecture: Authors in [106] propose an application layer solution for interoperability for IoT aiming to resolve the interoperability issue between different kinds of protocols (Bluetooth and UPnP). The key idea is to use device semantics provided by existing specifications and dynamically wrap them in their SOA middleware into semantic services.

SENSEI [111] is a SOA middleware for IoT. It includes a context model, context services, actuation tasks, and dynamic service composition of services. The main component of this middleware is the resource layer, which stands between the application layer and communication services layer. Resources in SENSEI use ontologies for their semantic modeling.

ubiSOAP [24] is a lightweight service oriented middleware that offers resource management and network level interoperability by supporting heterogeneous networking devices and technologies. Dynamic composition and instantiation of new services are facilitated by the semantically rich models and XML descriptions of sensors, actuators, and processing elements. The lack of context-awareness in ubiSOAP could be an issue, as this is key in the adaptive and autonomous behavior of the things.

SMArc [99] is an acronym for Semantic Middleware Architecture based on SOA. It focuses on smart city energy management for smart grid environments. A light ontology and a data representation under a specific format have been used in the implementation in order to guarantee that the interchanged data uses a representation format which will be common in the system, and therefore information will be easier to extract.

OM2M [6] is an advanced semantic middleware based on SOA architecture. It is a Machine-to-Machine service based on autonomic computing and semantic annotation to provide an inter-operable system to connect billions of devices, but they do not consider real-time analysis and full loosely coupled architecture. Authors proposed the IoT Ontology (IoT-O) [104] for the autonomic management of M2M systems. They extended SSN with four main modules: Acting

module to describe actuators, Lifecycle module to model state machines, Service module to describe services and finally the Energy module to represent the power consumption for appliances. However, this ontology does not consider the description of virtual sensors and cognitive processing of information.

Another example is the European project OpenIoT [68] that has developed an open-source middleware platform providing a “cloud-of-things”. OpenIoT aims to propose on-demand access to cloud-based IoT services for Internet-connected objects. Trying to use sensing as a service, OpenIoT architecture embeds the CUPUS middleware as a cloud-Based publish/subscribe processing engine and relies on SSN for sensors description. The observed data is stored as linked data and processed based on SPARQL queries which are continuously executed as data arrive. It can be viewed as a federation of several middleware projects interconnected with each other targeting applications for smart cities or campus and agriculture domain. OpenIoT could have been a good candidate to target IoT health applications, but its complexity due to the variety of middleware solutions under use can be a major drawback for the programmer.

The aforementioned middleware projects seek to tackle general IoT requirements without taking into account specific IoT application domains. They evaluated their proposals by deploying IoT sensors and testing the performance in some context however, they present their work as general and applicable to IoT applications. From the other side, Semantic middleware approaches have been proposed in a specific domain such as **healthcare**.

In this context LinkSmart and KASOM are SOA-based middleware for **medical applications**. LinkSmart is SOA middleware [108] developed for diabetes monitoring and therapy in the REACTION project. IT relies on a semantic model-driven architecture and enables the use of devices as services. The semantic description of devices is based on ontologies. KASOM [32] is a knowledge-Aware Service Oriented Middleware proposal for pervasive environments. Its architecture consists of offering services through registration, discovery, composition, and orchestration of services. Most of these services are established on complex reasoning mechanisms and protocols based on a contextual model, which represents a semantic description of low- and high-level resources of the WSN. Real-life implementations in hospital and health management show its potential in terms of response time, efficiency, and reliability. However, KASOM does not provide dynamic service composition in mobile and resource constrained IoT infrastructures because of predefined service composition rules provided by in-network agents.

From **AAL applications** perspective, openAAL [117] is an open source middleware for AAL applications, it relies on SOA architecture as a communication paradigm and on ontology description enabling service discovery. It defines a framework on top of the OSGi specification to facilitate integration and communication among services, including the context manager, procedural manager, and composer. CHOReOS [53] enables large scale choreographies or compositions of adaptable, QoS-aware, and heterogeneous services in IoT. It provides a scalable probabilistic thing-based service registration and discovery

to address scalability, interoperability, mobility, and adaptability. It uses ontologies for semantic enrichment over the architecture. The semantic thing based service compositions are transparently and automatically executable with no involvement from end-users, which is highly desirable in IoT, especially in M2M communications.

Semantic MoM-Based Architecture: Semantic MoM-based proposals have not been widely discussed in the literature regarding SOA-based proposals. However, new solutions have considered MoM and semantic description in their research work. Namely, SITRUS, EnTimid and xAAL middleware proposals. Bispo et al. propose SITRUS (Semantic Infrastructure for Wireless Sensor Networks) in [17]. Beside a MOM communication model, authors rely on ontology as semantic description processing module whose purpose is to generate a semantic database that provides the basis to decide whether a WSN node needs to be reconfigured or not.

EnTimid and xAAL have been proposed for **AAL applications**. EnTimid [83] is a MOM middleware for smart home monitoring. It relies on a component model for sensors description but it does not address messages or domain description in this contribution. xAAL [72] is another example of MOM middleware, it has been designed in the context of the PRECIOUS⁸ European project for AAL. Sensors schema model has been proposed in this work for sensors description and system components exchange Json messages between them.

4.2 Overview on Existing Semantic Middleware Platforms

The previously mentioned approaches have been presented in the literature as semantic middleware solutions. However, other middleware solutions have been introduced through IoT platforms. In [75, 79, 80] for instance, authors present a large review of 38 contemporary IoT platforms. They described each platform and classified them based on seven characteristics: support of heterogeneous devices, type of the platform, architecture's design, proprietary or open source, support of REST, data access control and service discovery.

As presented in Table 2, the second column shows if the platform supports heterogeneous devices or otherwise what kind of sensors it handles. We can see that most of the described projects support heterogeneous devices. Some solutions like Everyware needs a gateway to manage devices and there are solutions that support only one kind of sensors like Fosstrack that supports only RFID sensors.

We also deduce that in most cases, the platforms are provisioned from a cloud, either in the form of a Platform-as-a-Service (PaaS) or a Software-as-a-Service (SaaS). This type of platforms provides storage facilities, devices management, device connectivity, backup mechanisms or online support.

Table 2 also includes information about the openness of the platforms. Open-source platforms are considered more promising compared with the proprietary

⁸ <http://www.thepreciousproject.eu/>.

Table 2. Comparative table (Recap) of IoT Platforms [80]

Platforms	Support heterogeneous devices	of Type	Architecture	Open source	REST Data access control	Service discovery	Data description
AirVantage	Needs gateway	M2M PaaS	Cloud-based	Libraries only (Apache v2, MIT and Eclipse v1.0)	Yes	OAuth2	No n.a.
Arkessa	Yes	M2M PaaS	Cloud-based	No	n.a.	Facebook like privacy settings	No n.a.
ARM mbed	Embedded devices	de- M2M PaaS	Centralized/ Cloud-based	No	CoAP	User's choice	No n.a.
Carriots	Yes	PaaS	Cloud-based	No	Yes	Secured access	No JSON, XML
DeviceCloud	Yes	PaaS	Cloud-based	No	Yes	n.a.	No n.a.
EveryAware	Yes	Server	Centralized	No	Yes	4 levels	No JSON, XML
Everyware	Needs gateway	Paas	Cloud-based	No	Yes	n.a.	No JSON, XML
EvryThing	Yes	M2M PaaS	Centralized	No	Yes	Fine-grained	No n.a.
Exosite	Yes	Paas	Cloud-based	Libraries only (BSD license)	Yes	n.a.	No JSON
Fosstrack	RFID	Server	Centralized	No	No	Locally stored	No n.a.
GroveStreams	No	PaaS	Cloud-based	No	Yes	Role-based	No JSON
Hub-of-All-Things H.A.T	Home devices	PaaS	Decentralized	Yes	Yes	Locally stored	Yes n.a.
IoT-framework	Yes	Server	Centralized	Apache license 2.0	Yes	Locally stored	Yes n.a.
IFTTT	Yes	SaaS	Centralized	No	No	No storage	Limited n.a.
Kahvhub	Yes	Server	Centralized	Apache license 2.0	Yes	Locally stored	Yes n.a.
LinkSmart	Embedded devices	de- P2P	Decentralized	LGPLv3	No	Locally stored	Yes ontologies to discover devices
MyRobots	Robots	Robots PaaS	Cloud-based	No	Yes	2 levels	No JSON, XML
Niagara	Yes	M2M SaaS	Distributed	No	n.a.	n.a.	n.a. n.a.
Nimbits	Yes	Server	Centralized/ Cloud-based	Apache license 2.0	Yes	3 levels	No JSON
NinjaPlatform	Needs gateway	PaaS	Cloud-based	Open source hard-ware and Operating System	Yes	OAuth2	No JSON
Node-RED	Yes	Server	Centralized	Apache license 2.0	No	User-based privileges	No n.a.
OpenIoT	Yes	Hub	Decentralized	LGPLv3	No	User-based privileges	Yes ontologies to discover devices
OpenMTC	Yes	M2M client/server	Centralized/ Cloud-based	No	Yes	Secured access	No n.a.
OpenRemote	Home devices	Server	Centralized	Affero GNU Public License	Yes	Locally stored	No n.a.
Open.Sen.se	Ethernet enabled	PaaS / SaaS	Cloud-based	No	Yes	2 levels	Limited JSON
realTime.io	Needs gateway	PaaS	Cloud-based	No	Yes	Secured access	No JSON
SensorCloud	No	PaaS	Cloud-based	No	Yes	n.a.	No n.a.
SkySpark	No	SaaS	Centralized/ Cloud-based	No	Yes	n.a.	No n.a.
Swarm	Yes	PaaS	Cloud-based	Client is open source (unknown license)	Yes	n.a.	n.a. JSON
TempoDB	No	PaaS	Cloud-based	No	Yes	Secured access	No n.a.
TerraSwarm	Yes	OS	Decentralized	n.a.	n.a.	n.a.	Yes n.a.
The thing system	Home devices	Server	Centralized	M.I.T.	Yes	User's choice	No n.a.
Thing Broker	Yes	Server	Centralized	Yes	Yes	Locally stored	No n.a.
ThingsSpeak	Yes	Server	Centralized/ Cloud-based	GNU GPLv3	Yes	2 levels	Limited JSON, XML
ThingSquare	Embedded devices	de- Mesh	Cloud-based	Gateway firmware is open source	Yes	No	No n.a.
ThingWorx	Yes	M2M PaaS	Cloud-based	No	Yes	User-based privileges	Yes n.a.
WoTkit	Yes	PaaS	Cloud-based	No	Yes	Secured access	Yes n.a.
Xively	Yes	PaaS	Cloud-based	Libraries are open source (BSD 3-clause), platform is not	Yes	Secured access	Yes JSON, XML

alternatives because they are expected to enable the faster integration of new IoT solutions across the application domains. Among the platforms presented in the table, only 11 are open-source.

Moreover, it may be deduced from the table that only a few platforms do not have a REST API. This demonstrates that the current IoT services will tend to adopt the web of things paradigm [89]. We also deduce that only a few platforms have integrated some type of service discovery mechanisms. A comprehensive survey on discovery protocols for M2M communications can be found in [113].

The presented solutions have not been studied from a semantic perspective. For that aim, we added the last column in the table to show if a platform integrates semantic description or not. We also present the semantic data format when the semantic description is applied. As it is depicted in the table, few platforms have integrated semantic descriptions in their solution. JSON and XML are predominant used formats in both open and proprietary platforms.

Besides the before mentioned platforms, there are some open source solutions that have been developed in the context of home automation and AAL environments. We have studied these solutions and classified them in Table 2 taking into consideration how the semantic description in each solution is addressed.

- AllJoyn Lambda [112] relies on Bus-D architecture and offers an open source framework for the management of IoT environments. It can be seen as a software solution integrating AllJoyn in the Lambda architecture used for Big Data storage and analytics. This proposal offers a graphical user interface (GUI) for devices discovery and management, secure data transport between communication technologies (Wi-Fi, Bluetooth, etc.), interoperability between different OS and notifications interface that sends/receives human readable messages. A simple smart home case study has been tested using this approach. It consists on turning on/off devices at home. Semantic descriptions have not been considered in this approach.
- Kaa⁹ is a server-endpoints platform with the aim to create IoT applications including applications for healthcare, agriculture and smart city. It has been promoted as compatible virtually with any type of connected devices, sensors, and gateways. However, Kaa requires the integration of a specific microchip in the hardware of the IoT device which can be problematic for commercial sensors.
- Mango¹⁰ is a modular web-application framework. The main functionalities offered by this proposal are data collection, real time data monitoring, high performance database, logic and automation, security, cross platform, graphic dashboard and internal performance monitoring. It is based on a list of middleware and on an application that is compiled into a single HTTP server object. The middleware and applications are written in a functional style, which keeps everything modular. Semantic description is not provided.

⁹ <https://www.kaaproject.org>.

¹⁰ <https://github.com/paulbellamy/mango>.

- Nimbits¹¹ is an open source PaaS that can be used for hardware and software applications. It can be downloaded on platforms like raspberry PI, Amazon EC2 and Google App Engine. This solution has many key features including geo and time-stamped data processing, event and alerts triggering and offer a build provided for Google App Engine and Linux Systems but it does not offer semantic features.
- OpenRemote [65] is a software integration platform for residential and commercial building automation with the ambition to overcome the challenges of integration between many different protocols and solutions available for home automation. It relies on cloud-based design tools to offer user interface design, installation management and configuration. An Internet connection is only required for communication to systems outside the own network, or during configuration. Complex IoT applications with decision module programming are not supported in this platform neither are semantic descriptions.
- servIoTicy API [88] is an IoT-as-a-Service Data Management Platform. It provides multi-provider data stream processing capabilities on the cloud, a REST API, data analytics, advanced queries and multi-protocol support in a combination of advanced data-centric technologies. This work is still in early stages.
- OpenIoT [68] is a generic and open source middleware platform funded by the European Union. It offers open access to a wide range of technologies for Internet-connected sensors and other objects exposed as services with the ability to support large-scale deployments. It offers a friendly user interface and allows to link together different IoT devices and semantic web services.

Commercial IoT platforms have been put into light by technology companies such as Amazon and Microsoft. We summarize some of the industrial IoT platforms in this section.

- Predix¹² is a PaaS and cloud-based IoT platform made for mainstream sectors like aviation. The main three components are *Predix Machine* responsible for collecting data from industrial assets and pushing it to the *Predix cloud* which is a global, secure cloud infrastructure. *Predix Services* are used by developers to build, test, and run industrial IoT applications.
- IBM Watson¹³ for its part, employs speech and visual recognition, analyses the visual content of images and videos to understand their content.
- AWS from Amazon¹⁴ and Azure IoT Suite¹⁵ present successful IoT platforms in industrial domain. In addition to their private philosophy, industrial solutions do not consider semantic description in their proposals.

¹¹ <https://www.nimbits.com/>.

¹² <https://www.ge.com/digital/predix/industries>.

¹³ <https://www.ibm.com/watson/>.

¹⁴ <https://aws.amazon.com/>.

¹⁵ <https://azure.microsoft.com/en-us/suites/iot-suite/>.

4.3 Discussion

To sum up, a comparative table is presented in Table 3 with respect to the different perspectives of semantic modeling in IoT: sensors, messages and domain. This table summarizes the open source semantic middleware solutions cited in this section with respect to the two middleware architectures SOA and MOM. Finally, we provide the application domain that each solution has been proposed for.

Table 3. Comparative table (Recap) of Semantic Middleware solutions for IoT

Middleware projects	Architecture type		Semantic description			Application
	SOA	MOM	Sensors	Messages	Domain	
Song et al.	X		X			Sensors as services
OpenAAL	X		X			IT services, Living Lab
Sensei	X		X			IoT applications
LinkSmart	X		X			Diabetes monitoring and therapy
SM4ALL	X		X			AAL (elderly, disabled people)
EnTimid		X				AAL, Home automation
ubiSOAP	X		X			Ubiquitous networking
KASOM	X		X			Ubiquitous environments
CHOReOS	X		X	X		AAL, Home automation
SMArc	X		X	X		Smart city, Smart grid
universAAL	X					AAL, Home automation
SOPRANO	X		X			AAL, Home automation
OM2M	X		X	X		IoT applications
xAAL		X	X	X		AAL, Home automation
XGSN (OpenIoT)	X		X	X		OpenIoT, Smart city, Agriculture
SITRUS		X	X	X		Automatic reconfiguration of WSN
AllJoyn	X		X			AAL, Home automation
Kaa	X		X			AAL (Elderly, Disabled people)
Mango	X		X			Diabetes monitoring and therapy
OpenRemote	X		X			AAL, Home automation
servIoTicy API		X	X	X		AAL, Home automation

As depicted in this table, semantic IoT middleware proposals have been integrated in different domain applications, namely in AAL and home automation, monitoring of elderly and disabled people, smart cities, agriculture, and monitoring of vital and health signs. It is noticeable the predominant choice of SOA architecture as a middleware solution. Moreover, when semantic approaches are proposed they are mainly integrated for semantic descriptions of sensors. But there is no full description including sensors, sensor data and domain.

Moreover, the semantic description in SOA architectures has been proposed in order to describe services which refer to sensors specifications in IoT systems. For example, XML schema has been proposed in SM4ALL [25] for sensors description and used as a contract for applications service description.

Other semantic description techniques can also be noticed like model driven architecture for devices description in [62] and Linksmart¹⁶. Lately, ontologies have gained sufficient importance for knowledge and sensors description as it is presented in SOPRANO [105], Sensei [91] OpenIoT, SMArc, UBIWARE and OM2M. Another important aspect of this comparison study is the lack of messages and domain description in these propositions, we can see that two ontology-based solutions OM2M and XGSN/OpenIoT have considered OWL format for exchanging messages.

From the other side, only a few attempts address the semantic topic in their proposals and most of these attempts rely on existing SOA-based solutions. So we end up with more semantic SOA-based than MOM-based architectures for IoT. But as previously mentioned, to provide a loose coupling and information centric solution, SOA-based solutions are not so relevant. Therefore, MOM in the past few years has been a topic of interest for IoT researcher and some semantic MOM solutions have been proposed. Entimid, XAAL and SITRUS are MOM-based middleware solutions that have addressed semantic representation in their solution. However, these solutions do not offer a full description model for IoT as it is shown in the table.

5 Conclusion and Ongoing Challenges

IoT has been gradually bringing changes in the healthcare domain. There is innumerable usefulness of IoT applications in enhancing the quality of life of elderly and in helping medical staff. Though IoT has abundant benefits, there are some challenges related to the heterogeneous components in an IoT system that should communicate and exchange data. In this book chapter we have presented several paradigms used to answer these challenges. Some projects have proposed to tackle the interoperability issue from a technological perspective by relying on middleware solutions that promote the interoperability and the manageability of sensors in an IoT system. From a semantic perspective, ontologies have been used to promote the semantic interoperability in IoT systems and to ensure the homogeneity of data formats. The coupling of both technologies has been found as a promising solution for IoT environments. Hence, semantic middleware architectures have been proposed as a complete IoT solution.

Although the existing semantic middleware proposals address many challenges and requirements regarding the interoperability in IoT systems, there are still some open research challenges related to the scalability and real-time reasoning. Using ontologies affects these two requirements as parsing, storing, inferencing and querying of/over OWL/RDF data as well as communicating such data take longer time than working with simple native data. Furthermore, ontologies present a high computational cost and require domain knowledge and expertise leading developers to avoid its use. Furthermore, the complexity related to the diversity of libraries to use and the complexity of the programming environment, make its use more complex. Providing simple software API usable in

¹⁶ <http://hydramiddleware.eu/news.php>.

various application domains, to alleviate the tasks of software developers who decide to rely on semantic middleware architecture would be an open challenge to investigate. Another challenge that could be addressed in the research area is related to ontologies for sensors and domain description. As pointed out in most projects, these two aspects are described separately. Providing a complete ontology that is able to describe both domain and sensors in IoT is still a challenge to tackle. From the other side, using MoM approaches with semantic description in IoT is still in an early stage. Extensive researches in this context help invention of promising interoperable and scalable IoT architecture that adopt the Web of Things concepts as described in [119].

Web of Thing (WoT) [50] paradigm has been introduced as one of the major solutions for interoperability issue in IoT. It is an improvement of IoT allowing an easier way for IoT applications to build upon smart things. The WoT concept relies on the connectivity service of IoT and easy access to sensors in order to create applications exploiting the IoT data [48]. It can be seen as an evolution of the Internet of things where all components share their information and collaborate to generate advanced knowledge such as wisdom. Moreover, using semantic description allows data contextualization for optimized data stream discovery, indexing and querying. Moreover, new research projects are shifting to the semantic web of things [90] where data can be integrated with data and services available in other information systems. This flexibility eases the production of novel applications and services that are based on the state of the real world. It also supports autonomous semantic reasoning and decision making mechanisms to provide higher-level actionable knowledge from low-level sensor data [39]. Hence, performing semantic reasoning is linked to the ability to define a description model of sensors observations. The WoT is an open research topic to improve and investigate by testing in several IoT environments such as Healthcare.

References

1. iDigi device cloud. <https://www.digi.com/>
2. Internet of Things thingspeak service. <https://thingspeak.com/>
3. The pachube feed cloud service. <https://www.xively.com/>
4. Aberer, K., Hauswirth, M., Salehi, A.: A middleware for fast and flexible sensor network deployment. In: Proceedings of the 32nd International Conference on Very Large Data Bases, pp. 1199–1202. VLDB Endowment (2006)
5. Alam, S., Chowdhury, M.M., Noll, J.: SenaaS: an event-driven sensor virtualization approach for Internet of Things cloud. In: 2010 IEEE International Conference on Networked Embedded Systems for Enterprise Applications (NESEA), pp. 1–6. IEEE (2010)
6. Alaya, M.B., Banouar, Y., Monteil, T., Chassot, C., Drira, K.: OM2M: extensible ETSI-compliant M2M service platform with self-configuration capability. *Procedia Comput. Sci.* **32**, 1079–1086 (2014). <https://doi.org/10.1016/j.procs.2014.05.536>
7. Alaya, M.B., Medjiah, S., Monteil, T., Drira, K.: Towards semantic data interoperability in oneM2M standard. *IEEE Commun. Mag.* **53**(12), 35 (2015)

8. Alex, H., Kumar, M., Shirazi, B.: MidFusion: an adaptive middleware for information fusion in sensor network applications. *Inf. Fusion* **9**(3), 332–343 (2008)
9. Anastasi, G.F., Bini, E., Romano, A., Lipari, G.: A service-oriented architecture for QoS configuration and management of wireless sensor networks. In: 2010 IEEE Conference on Emerging Technologies and Factory Automation (ETFA), pp. 1–8. IEEE (2010)
10. Avilés-López, E., García-Macías, J.A.: TinySOA: a service-oriented architecture for wireless sensor networks. *Serv. Oriented Comput. Appl.* **3**(2), 99–108 (2009)
11. Bandyopadhyay, S., Sengupta, M., Maiti, S., Dutta, S.: Role of middleware for Internet of Things: a study. *Int. J. Comput. Sci. Eng. Surv.* **2**(3), 94–105 (2011)
12. Bazzani, M., Conzon, D., Scalera, A., Spirito, M.A., Trainito, C.I.: Enabling the IoT paradigm in e-health solutions through the virtus middleware. In: 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), pp. 1954–1959. IEEE (2012)
13. Benatallah, B., Motahari Nezhad, H.R.: Service oriented architecture: overview and directions. In: Börger, E., Cisternino, A. (eds.) *Advances in Software Engineering*. LNCS, vol. 5316, pp. 116–130. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-89762-0_4
14. Benazzouz, Y., Parissis, I., et al.: A fault fuzzy-ontology for large scale fault-tolerant wireless sensor networks. *Procedia Comput. Sci.* **35**, 203–212 (2014)
15. Bendadouche, R., Roussey, C., Sousa, G.D., Chanet, J.P., Hou, K.M.: Extension of the semantic sensor network ontology for wireless sensor networks: the stimulus-WSNnode-communication pattern. In: 5th International Workshop on Semantic Sensor Networks in Conjunction with the 11th International Semantic Web Conference (ISWC) (November 2012, Boston, US) (2013)
16. Bermudez, L., Delory, E., O'Reilly, T., del Rio Fernandez, J.: Ocean observing systems demystified. In: *MTS/IEEE Biloxi-Marine Technology for Our Future: Global and Local Challenges*, OCEANS 2009, pp. 1–7. IEEE (2009)
17. Bispo, K.A., Rosa, N.S., Cunha, P.R.: Citrus: semantic infrastructure for wireless sensor networks. *Sensors* **15**(11), 27436–27469 (2015)
18. Botts, M.: OGC implementation specification 07–000: OpenGIS sensor model language (SensorML)-open geospatial consortium. Technical report (2007)
19. Botts, M., Percivall, G., Reed, C., Davidson, J.: OGC® sensor web enablement: overview and high level architecture. In: Nittel, S., Labrinidis, A., Stefanidis, A. (eds.) *GSN 2006*. LNCS, vol. 4540, pp. 175–190. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-79996-2_10
20. Brandt, P., et al.: Semantic interoperability in sensor applications making sense of sensor data. In: 2013 IEEE Symposium on Computational Intelligence in Healthcare and e-health (CICARE), pp. 34–41. IEEE (2013)
21. Brouwers, N., Langendoen, K.: Pogo, a middleware for mobile phone sensing. In: Narasimhan, P., Triantafyllou, P. (eds.) *Middleware 2012*. LNCS, vol. 7662, pp. 21–40. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-35170-9_2
22. Calder, M., Morris, R.A., Peri, F.: Machine reasoning about anomalous sensor data. *Ecol. Informat.* **5**(1), 9–18 (2008)
23. Cantais, J., Dominguez, D., Gigante, V., Laera, L., Tamma, V.: An example of food ontology for diabetes control. In: *Proceedings of the International Semantic Web Conference 2005 Workshop on Ontology Patterns for the Semantic Web*, pp. 1–9 (2005)
24. Caporuscio, M., Raverdy, P.G., Issarny, V.: ubiSOAP: a service-oriented middleware for ubiquitous networking. *IEEE Trans. Serv. Comput.* **5**(1), 86–98 (2012)

25. Catarci, T., et al.: Service composition and advanced user interfaces in the home of tomorrow: the SM4All approach. In: Gabrielli, S., Elias, D., Kahol, K. (eds.) AMBI-SYS 2011. LNICST, vol. 70, pp. 12–19. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23902-1_2
26. Chen, L., Nugent, C., Okeyo, G.: An ontology-based hybrid approach to activity modeling for smart homes. *IEEE Trans. Hum.-Mach. Syst.* **44**(1), 92–105 (2014). <https://doi.org/10.1109/thms.2013.2293714>
27. Chen, L., Nugent, C.D., Wang, H.: A knowledge-driven approach to activity recognition in smart homes. *IEEE Trans. Knowl. Data Eng.* **24**(6), 961–974 (2012). <https://doi.org/10.1109/tkde.2011.51>
28. Compton, M., et al.: The SSN ontology of the W3C semantic sensor network incubator group. *Web Semant.: Sci. Serv. Agents World Wide Web* **17**, 25–32 (2012). <https://doi.org/10.1016/j.websem.2012.05.003>
29. Compton, M., Henson, C., Lefort, L., Neuhaus, H., Sheth, A.: A survey of the semantic specification of sensors. In: Proceedings of the 2nd International Conference on Semantic Sensor Networks, vol. 522, pp. 17–32. CEUR-WS.org (2009)
30. Corchado, J.M., Bajo, J., Tapia, D.I., Abraham, A.: Using heterogeneous wireless sensor networks in a telemonitoring system for healthcare. *IEEE Trans. Inf. Technol. Biomed.* **14**(2), 234–240 (2010)
31. Coronato, A.: Uranus: a middleware architecture for dependable AAL and vital signs monitoring applications. *Sensors* **12**(3), 3145–3161 (2012)
32. Corredor, I., Martínez, J.F., Familiar, M.S., López, L.: Knowledge-aware and service-oriented middleware for deploying pervasive services. *J. Netw. Comput. Appl.* **35**(2), 562–576 (2012)
33. Costa, P., Mottola, L., Murphy, A.L., Picco, G.P.: TeenyLIME: transiently shared tuple space middleware for wireless sensor networks. In: Proceedings of the International Workshop on Middleware for Sensor Networks, pp. 43–48. ACM (2006)
34. Curino, C., Giani, M., Giorgetta, M., Giusti, A., Murphy, A.L., Picco, G.P.: TinyLIME: bridging mobile and sensor networks through middleware. In: Third IEEE International Conference on Pervasive Computing and Communications, PerCom 2005, pp. 61–72. IEEE (2005)
35. De, S., Barnaghi, P., Bauer, M., Meissner, S.: Service modelling for the Internet of Things. In: 2011 Federated Conference on Computer Science and Information Systems (FedCSIS), pp. 949–955. IEEE (2011)
36. Delamer, I.M., Lastra, J.L.M.: Service-oriented architecture for distributed publish/subscribe middleware in electronics production. *IEEE Trans. Ind. Informat.* **2**(4), 281–294 (2006)
37. D’Elia, A., Roffia, L., Zamagni, G., Vergari, F., Toninelli, A., Bellavista, P.: Smart applications for the maintenance of large buildings: how to achieve ontology-based interoperability at the information level. In: 2010 IEEE Symposium on Computers and Communications (ISCC), pp. 1–6. IEEE (2010)
38. Delicato, F.C., Pires, P.F., Batista, T.: Middleware Solutions for the Internet of Things. Springer, Heidelberg (2013). <https://doi.org/10.1007/978-1-4471-5481-5>
39. Desai, P., Sheth, A., Anantharam, P.: Semantic gateway as a service architecture for IoT interoperability. In: Mobile Services (MS), pp. 313–319. IEEE (2015)
40. Díaz-Rodríguez, N., Härmä, A., Helouai, R., Huitzil, I., Bobillo, F., Straccia, U.: Couch potato or gym addict? Semantic lifestyle profiling with wearables and knowledge graphs. In: Proceedings of the 6th Workshop on Automated Knowledge Base Construction (AKBC 2017), Long Beach (USA) (2017)

41. Díaz-Rodríguez, N., Cadahía, O.L., Cuéllar, M.P., Lilius, J., Calvo-Flores, M.D.: Handling real-world context awareness, uncertainty and vagueness in real-time human activity tracking and recognition with a fuzzy ontology-based hybrid method. *Sensors* **14**(10), 18131–18171 (2014)
42. Donnelly, K.: SNOMED-CT: the advanced terminology and coding system for e-health. *Stud. Health Technol. Informat.* **121**, 279 (2006)
43. Doukas, C., Maglogiannis, I.: Bringing IoT and cloud computing towards pervasive healthcare. In: 2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), pp. 922–926. IEEE (2012)
44. Duarte, P.H., Faina, L.F., Camargos, L.J., de Paula, L.B., Pasquini, R.: An architecture for monitoring and improving public transportation systems. In: 2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA), pp. 871–878. IEEE (2016)
45. Evensen, P., Meling, H.: Sensewrap: a service oriented middleware with sensor virtualization and self-configuration. In: 2009 5th International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), pp. 261–266. IEEE (2009)
46. Fok, C.L., Roman, G.C., Lu, C.: Agilla: a mobile agent middleware for self-adaptive wireless sensor networks. *ACM Trans. Auton. Adapt. Syst. (TAAS)* **4**(3), 16 (2009)
47. Gomez, M., et al.: An ontology-centric approach to sensor-mission assignment. In: Gangemi, A., Euzenat, J. (eds.) EKAW 2008. LNCS (LNAI), vol. 5268, pp. 347–363. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-87696-0_30
48. Guinard, D., Trifa, V.: Towards the web of things: web mashups for embedded devices. In: Proceedings of WWW (International World Wide Web Conferences) Workshop on Mashups, Enterprise Mashups and Lightweight Composition on the Web (MEM 2009), Madrid, Spain, vol. 15 (2009)
49. Guinard, D., Trifa, V., Karnouskos, S., Spiess, P., Savio, D.: Interacting with the SOA-based Internet of Things: discovery, query, selection, and on-demand provisioning of web services. *IEEE Trans. Serv. Comput.* **3**(3), 223–235 (2010)
50. Guinard, D., Trifa, V., Mattern, F., Wilde, E.: From the Internet of Things to the web of things: resource-oriented architecture and best practices. In: Uckelmann, D., Harrison, M., Michahelles, F. (eds.) Architecting the Internet of Things, pp. 97–129. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-19157-2_5
51. Gyrard, A., Bonnet, C., Boudaoud, K.: Enrich machine-to-machine data with semantic web technologies for cross-domain applications. In: 2014 IEEE World Forum on Internet of Things (WF-IoT), pp. 559–564. IEEE (2014)
52. Gyrard, A., Datta, S.K., Bonnet, C., Boudaoud, K.: Standardizing generic cross-domain applications in Internet of Things. In: Globecom Workshops (GC Wkshps), pp. 589–594. IEEE (2014)
53. Hamida, A.B., et al.: Integrated choreos middleware-enabling large-scale, QoS-aware adaptive choreographies (2013)
54. Hassan, M.M., Albakr, H.S., Al-Dossari, H.: A cloud-assisted Internet of Things framework for pervasive healthcare in smart city environment. In: Proceedings of the 1st International Workshop on Emerging Multimedia Applications and Services for Smart Cities, pp. 9–13. ACM (2014)
55. Heinzelman, W.B., Murphy, A.L., Carvalho, H.S., Perillo, M.A.: Middleware to support sensor network applications. *IEEE Netw.* **18**(1), 6–14 (2004)

56. Hendi, H.I., Ahmad, A., Bouneffa, M., Fonlupt, C.: Ontology based reasoning for solving passenger train optimization problem. In: Al-Sadeq International Conference on Multidisciplinary in IT and Communication Science and Applications (AIC-MITCSA), pp. 1–6. IEEE (2016)
57. Herzog, A., Jacobi, D., Buchmann, A.: A3ME-an agent-based middleware approach for mixed mode environments. In: The Second International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, UBIComm 2008, pp. 191–196. IEEE (2008)
58. Hong, K., et al.: TinyVM: an energy-efficient execution infrastructure for sensor networks. *Softw.: Pract. Exp.* **42**(10), 1193–1209 (2012)
59. Hoque, E., Stankovic, J.: AALO: activity recognition in smart homes using active learning in the presence of overlapped activities. In: 2012 6th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth), pp. 139–146. IEEE (2012)
60. Huebscher, M.C., McCann, J.A.: Adaptive middleware for context-aware applications in smart-homes. In: Proceedings of the 2nd Workshop on Middleware for Pervasive and Ad-Hoc Computing, pp. 111–116. ACM (2004)
61. Ibrahim, A., Carrez, F., Moessner, K.: Spatio-temporal model for role assignment in wireless sensor networks. In: Proceedings of the 2013 19th European Wireless Conference (EW), pp. 1–6. VDE (2013)
62. Janse, M.D.: Amigo Final Report, pp. 1–42, September 2008
63. Jonquet, C., et al.: AgroPortal: an open repository of ontologies and vocabularies for agriculture and nutrition data. In: GODAN Summit (2016)
64. Junnila, S., et al.: Wireless, multipurpose in-home health monitoring platform: two case trials. *IEEE Trans. Inf. Technol. Biomed.* **14**(2), 447–455 (2010)
65. Jyothi, T.: Open source middleware for Internet of Things. *Int. J. Innov. Res. Electr. Electron. Instrum. Control Eng.* **4**(11), 71–75 (2016). <http://www.openremote.com/>
66. Kang, P., Borcea, C., Xu, G., Saxena, A., Kremer, U., Iftode, L.: Smart messages: a distributed computing platform for networks of embedded systems. *Comput. J.* **47**(4), 475–494 (2004)
67. Kibbe, W.A., et al.: Disease ontology 2015 update: an expanded and updated database of human diseases for linking biomedical knowledge through disease data. *Nucleic Acids Res.* **43**, D1071–D1078 (2014). <https://doi.org/10.1093/nar/gku1011>
68. Kim, J., Lee, J.W.: OpenIoT: an open service framework for the Internet of Things. In: 2014 IEEE World Forum on Internet of Things (WF-IoT). Institute of Electrical and Electronics Engineers (IEEE), March 2014. <https://doi.org/10.1109/wf-iot.2014.6803126>
69. Kim, J., Kim, J., Lee, D., Chung, K.Y.: Ontology driven interactive healthcare with wearable sensors. *Multimedia Tools Appl.* **71**(2), 827–841 (2014)
70. Lasierra, N., Alesanco, A., Guillén, S., García, J.: A three stage ontology-driven solution to provide personalized care to chronic patients at home. *J. Biomed. Inform.* **46**(3), 516–529 (2013)
71. Liu, T., Martonosi, M.: Impala: a middleware system for managing autonomic, parallel sensor systems. In: ACM SIGPLAN Notices, vol. 38, pp. 107–118. ACM (2003)
72. Lohr, C., Tanguy, P., Kerdreux, J.: xAAL: a distributed infrastructure for heterogeneous ambient devices. *J. Intell. Syst.* **24**(3) (2015). <https://doi.org/10.1515/jisys-2014-0144>

73. Mamei, M., Zambonelli, F.: Field-Based Coordination for Pervasive Multiagent Systems. Springer, Heidelberg (2006). <https://doi.org/10.1007/3-540-27969-5>
74. Marques, I.L., Ronan, J., Rosa, N.S.: TinyReef: a register-based virtual machine for wireless sensor networks. In: Sensors, pp. 1423–1426. IEEE (2009)
75. Mazhelis, O., Tyrvaïnen, P.: A framework for evaluating Internet-of-Things platforms: application provider viewpoint. In: 2014 IEEE World Forum on Internet of Things (WF-IoT), pp. 147–152. IEEE (2014)
76. Meditskos, G., Dasiopoulou, S., Kompatsiaris, I.: MetaQ: a knowledge-driven framework for context-aware activity recognition combining SPARQL and OWL 2 activity patterns. *Pervasive Mob. Comput.* **25**, 104–124 (2016). <https://doi.org/10.1016/j.pmcj.2015.01.007>
77. Megalingam, R.K., Pocklassery, G., Jayakrishnan, V., Mourya, G., Thulasi, A.A.: Smartphone based continuous monitoring system for home-bound elders and patients. In: 2014 International Conference on Communications and Signal Processing (ICCSP), pp. 1173–1177. IEEE (2014)
78. Miluzzo, E., et al.: Sensing meets mobile social networks: the design, implementation and evaluation of the CenceME application. In: Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems, pp. 337–350. ACM (2008)
79. Mineraud, J., Mazhelis, O., Su, X., Tarkoma, S.: Contemporary Internet of Things platforms. arXiv preprint [arXiv:1501.07438](https://arxiv.org/abs/1501.07438) (2015)
80. Mineraud, J., Mazhelis, O., Su, X., Tarkoma, S.: A gap analysis of Internet-of-Things platforms. *Comput. Commun.* **89**, 5–16 (2016)
81. Mohamed, N., Al-Jaroodi, J.: Service-oriented middleware approaches for wireless sensor networks. In: 2011 44th Hawaii International Conference on System Sciences (HICSS), pp. 1–9. IEEE (2011)
82. Mottola, L., Murphy, A.L., Picco, G.P.: Pervasive games in a mote-enabled virtual world using tuple space middleware. In: Proceedings of 5th ACM SIGCOMM Workshop on Network and System Support for Games, p. 29. ACM (2006)
83. Nain, G.: EnTiMid: Un modèle de composants pour intégrer des objets communicants dans des applications à base de services. Ph.D. thesis, Université Rennes 1 (2011)
84. Neuhaus, H., Compton, M.: The semantic sensor network ontology. In: AGILE Workshop on Challenges in Geospatial Data Harmonisation, Hannover, Germany, pp. 1–33 (2009)
85. Osello, A., et al.: Energy saving in existing buildings by an intelligent use of interoperable ICTs. *Energy Eff.* **6**(4), 707–723 (2013)
86. Palumbo, F., Barsocchi, P., Furfari, F., Ferro, E.: AAL middleware infrastructure for green bed activity monitoring. *J. Sens.* **2013** (2013)
87. Perera, C., Jayaraman, P.P., Zaslavsky, A., Georgakopoulos, D., Christen, P.: Mosden: an Internet of Things middleware for resource constrained mobile devices. In: 2014 47th Hawaii International Conference on System Sciences (HICSS), pp. 1053–1062. IEEE (2014)
88. Pérez, J.L., Carrera, D.: Performance characterization of the servioticy API: an IoT-as-a-service data management platform. In: 2015 IEEE First International Conference on Big Data Computing Service and Applications (BigDataService), pp. 62–71. IEEE (2015)
89. Pérez, J.L., Villalba, Á., Carrera, D., Larizgoitia, I., Trifa, V.: The compose API for the Internet of Things. In: Proceedings of the 23rd International Conference on World Wide Web, pp. 971–976. ACM (2014)
90. Pfisterer, D., et al.: SPITFIRE: toward a semantic web of things. *IEEE Commun. Mag.* **49**(11), 40–48 (2011)

91. Presser, M., Barnaghi, P.M., Eurich, M., Villalonga, C.: The SENSEI project: integrating the physical world with the digital world of the network of the future. *IEEE Commun. Mag.* **47**(4), 1–4 (2009)
92. Radhika, J., Malarvizhi, S.: Middleware approaches for wireless sensor networks: an overview. *Int. J. Comput. Sci. Issues (IJCSI)* **9**(3), 224–229 (2012)
93. Raychoudhury, V., Cao, J., Kumar, M., Zhang, D.: Middleware for pervasive computing: a survey. *Pervasive Mob. Comput.* **9**(2), 177–200 (2013)
94. Razzaque, M.A., Milojevic-Jevric, M., Palade, A., Clarke, S.: Middleware for Internet of Things: a survey. *IEEE Internet of Things J.* **3**(1), 70–95 (2016). <https://doi.org/10.1109/jiot.2015.2498900>
95. Riboni, D., Bettini, C.: OWL 2 modeling and reasoning with complex human activities. *Pervasive Mob. Comput.* **7**(3), 379–395 (2011). <https://doi.org/10.1016/j.pmcj.2011.02.001>
96. Roda, F., Musulin, E.: An ontology-based framework to support intelligent data analysis of sensor measurements. *Expert Syst. Appl.* **41**(17), 7914–7926 (2014)
97. Rodríguez, N.D., Cuéllar, M.P., Lilius, J., Calvo-Flores, M.D.: A fuzzy ontology for semantic modelling and recognition of human behaviour. *Knowl.-Based Syst.* **66**, 46–60 (2014)
98. Rodríguez, N.D., Cuéllar, M.P., Lilius, J., Calvo-Flores, M.D.: A survey on ontologies for human behavior recognition. *ACM Comput. Surv. (CSUR)* **46**(4), 43 (2014)
99. Rodríguez-Molina, J., Martínez, J.F., Castillejo, P., de Diego, R.: SMArc: a proposal for a smart, semantic middleware architecture focused on Smart City energy management. *Int. J. Distrib. Sens. Netw.* **2013** (2013)
100. Rueda, C., Bermudez, L., Fredericks, J.: The MMI ontology registry and repository: a portal for marine metadata interoperability. In: *MTS/IEEE Biloxi-Marine Technology for Our Future: Global and Local Challenges, OCEANS 2009*, pp. 1–6. IEEE (2009)
101. Schlenoff, C., Hong, T., Liu, C., Eastman, R., Foufou, S.: A literature review of sensor ontologies for manufacturing applications. In: *2013 IEEE International Symposium on Robotic and Sensors Environments (ROSE)*, pp. 96–101. IEEE (2013)
102. Scott, H.: Energy ontologies: wind, biomass, and fossil transportation. *Humanities* **5**(2), 37 (2016)
103. Seeger, C., Van Laerhoven, K., Buchmann, A.: MyHealthAssistant: an event-driven middleware for multiple medical applications on a smartphone-mediated body sensor network. *IEEE J. Biomed. Health Informat.* **19**(2), 752–760 (2015)
104. Seydoux, N., Drira, K., Hernandez, N., Monteil, T.: IoT-O, a core-domain IoT ontology to represent connected devices networks. In: Blomqvist, E., Ciancarini, P., Poggi, F., Vitali, F. (eds.) *EKAW 2016. LNCS (LNAI)*, vol. 10024, pp. 561–576. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-49004-5_36
105. Sixsmith, A., et al.: SOPRANO – an ambient assisted living system for supporting older people at home. In: Mokhtari, M., Khalil, I., Bauchet, J., Zhang, D., Nugent, C. (eds.) *ICOST 2009. LNCS*, vol. 5597, pp. 233–236. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-02868-7_30
106. Song, Z., Cárdenas, A.A., Masuoka, R.: Semantic middleware for the Internet of Things. In: *Internet of Things (IOT)*, pp. 1–8. IEEE (2010)
107. Souza, A.M., Amazonas, J.R.: A novel smart home application using an Internet of Things middleware. In: *Proceedings of 2013 European Conference on Smart Objects, Systems and Technologies (SmartSysTech)*, pp. 1–7. VDE (2013)

108. Spanakis, E.G., et al.: Diabetes management using modern information and communication technologies and new care models. *Interact. J. Med. Res.* **1**(2) (2012)
109. Su, X., Zhang, H., Riekkki, J., Keränen, A., Nurminen, J.K., Du, L.: Connecting iot sensors to knowledge-based systems by transforming SenML to RDF. *Procedia Comput. Sci.* **32**, 215–222 (2014)
110. Triantafyllidis, A.K., Koutkias, V.G., Chouvarda, I., Maglaveras, N.: A pervasive health system integrating patient monitoring, status logging, and social sharing. *IEEE J. Biomed. Health Informat.* **17**(1), 30–37 (2013)
111. Tsiatsis, V., et al.: The SENSEI real world internet architecture (2010)
112. Villari, M., Celesti, A., Fazio, M., Puliafito, A.: AllJoyn lambda: an architecture for the management of smart environments in IoT. In: 2014 International Conference on Smart Computing Workshops (SMARTCOMP Workshops), pp. 9–14. IEEE (2014)
113. Villaverde, B.C., de Paz Alberola, R., Jara, A.J., Fedor, S., Das, S.K., Pesch, D.: Service discovery protocols for constrained machine-to-machine communications. *IEEE Commun. Surv. Tutor.* **16**(1), 41–60 (2014)
114. Waluyo, A.B., Ying, S., Pek, I., Wu, J.K.: Middleware for wireless medical body area network. In: Biomedical Circuits and Systems Conference, BIOCAS 2007, pp. 183–186. IEEE (2007)
115. Warriach, E.U., Kaldeli, E., Lazovik, A., Aiello, M.: An interplatform service-oriented middleware for the smart home. *Int. J. Smart Home* **7**(1), 115–141 (2013)
116. Witt, K.J., et al.: Enabling sensor webs by utilizing SWAMO for autonomous operations. In: 8th NASA Earth Science Technology Conference, pp. 263–270 (2008)
117. Wolf, P., et al.: openAAL - the open source middleware for ambient-assisted living (AAL) (2010)
118. Ye, J., Stevenson, G., Dobson, S.: KCAR: a knowledge-driven approach for concurrent activity recognition. *Pervasive Mob. Comput.* **19**, 47–70 (2015)
119. Zgheib, R.: SeMoM: a semantic middleware for IoT healthcare applications. Ph.D. thesis, Université de Toulouse, Université Toulouse III-Paul Sabatier (2017)
120. Zhu, N., et al.: Bridging e-health and the Internet of Things: the sphere project. *IEEE Intell. Syst.* **30**(4), 39–46 (2015)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

