An Ontology-based Approach to Facilitate Semantic Interoperability of Context-Aware Systems

Hamed Barangi
MDSE Research Group, Department of Software Engineering, University of Isfahan, Isfahan, Iran
hbarangi@eng.ui.ac.ir

Shekoufeh Kolahdouz Rahimi
MDSE Research Group, Department of Software Engineering, University of Isfahan, Isfahan, Iran, and Sochool of Arts, University of Roehampton, London, UK
sh.rahimi@eng.ui.ac.ir

Bahman Zamani
MDSE Research Group, Department of Software Engineering, University of Isfahan, Isfahan, Iran
zamani@eng.ui.ac.ir

Hossein Moradi
Department of Computer Engineering, Birjand University of Technology, Birjand, Iran
hsmoradi@birjandut.ac.ir

***Abstract*— Semantic interoperability is one of the most critical challenges for software developers while integrating two or more context-aware systems. In such circumstances, it is essential to understand the meaning and interpretation of various contexts in different business domains and to align context ontologies together. Our investigations reveal that existing works poorly address these requirements. To fill this gap, this paper proposes the Ontology as a Service (OaaS) approach to facilitate the semantic interoperability of context-aware systems. In the proposed solution, the complexities of semantic interoperability are resolved and handled by a standalone ontology service, which can be easily reused and consumed by different ontology service consumers and brokers. The proposed ontology service includes several ontology repositories, a web service that positions a context concept in the existing ontologies and another web service that maps the relationship between existing context concepts. We evaluated our approach with a case study that resolves three semantic differences between two IoT applications originating from heterogeneous domains of smart home and health environments.**

Keywords— Ontology, Ontology as a Service, Semantic Interoperability, Context-Aware Systems

I. Introduction

The Internet of Things (IoT) is a system of interrelated devices connected to a network that collects and shares data. Context-aware systems are deployed in IoT environments, and adapt their operations actively and independently to the current context, and provide suitable services for the user and applications without user intervention [1].

Despite the heterogeneity, context-aware systems must be able to interact with different devices to complete their tasks and actions according to the current context [2]. In addition to heterogeneity in a specific domain, some applications need to exchange context information in different domains. In these cases, we are faced with the heterogeneity of modeling, storage, and context reasoning methods. The components of a context-aware system must be interoperable with other components and systems. Various aspects of heterogeneity have made interoperability one of the most significant development challenges in context-aware systems [3].

Interoperability consists of diverse aspects, including technical, syntactic, and semantic interoperability [4]. Semantic interoperability refers to the system's ability to recognize the meaning of exchanged data information and interpretation of content. Semantic interoperability in context-aware systems is crucial due to the need to understand contexts and perform context reasoning [2]. As described in the following, during data exchange between context-aware systems, three differences are considered in this article:

- The systems may have different context modeling techniques. Hence, there will be differences in common understanding between systems; for example, The context is modeled in one system with the ontology technique and in the other system with the markup-schema technique.
- The names of attributes used by sensors may be different; for example, the word “Location” may be used in one system and “Loc” in another system to refer to the location attribute.
- The third semantic difference is attribute content, which refers to the value of attributes; for example, the temperature attribute in one domain may refer to the ambient temperature, but in another, it refers to the body temperature.

Researchers have recently suggested various solutions and tools to achieve different types of interoperability in IoT environments. One of the critical solutions to solve the complexities of semantic interoperability in context-aware systems is to apply ontologies [5]. Ontology specifies the entities (concepts), characteristics, and the relationship between them in a specific domain. Supporting ontology is one of the critical features of context-aware systems that helps systems reach a common understanding of the meaning of the contexts. Achieving this understanding is necessary for the context modeling and reasoning stages of the context life cycle [6].

The growth of the number of sensors and volume of data increases the number of entities and relationships defined in ontologies. As a result, processing complex ontologies, which use many entities and relationships, is time-consuming. This performance challenge is more critical in environments requiring real-time response [7]. Many researchers suggested reducing the number of entities and relationships in an ontology, known as the lightweight ontology design [8][9]. Note that leveraging lightweight ontology design in the growing IoT environments cannot solve the performance challenge of ontologies since IoT devices are inherently limited in terms of memory consumption and processing power.

Ontology as a Service (OaaS) is a new approach that can resolve the performance challenge of applying ontology in IoT environments. OaaS can be used to facilitate the semantic interoperability of different context-aware systems. OaaS separates the consumer of the ontology service from its provider. OaaS completely shifts the complexity of semantic interoperability from the consumer side to the intermediate broker or the provider side.

In this paper, in addition to a lightweight ontology, a new OaaS architecture is proposed as a solution to facilitate the semantic interoperability of context-aware systems. Finally, a practical example is used to provide a comprehensive explain the proposed method in this research.

The paper is organized as follows: Section 2 presents a summary of related work in semantic interoperability between context-aware systems, section 3 discusses a motivational example of semantic interoperability challenges of context-aware systems, and section 4 explains the proposed OaaS solution. Section 5 evaluates the proposed solution, and finally, section 6 concludes and describes future work.

II. RELATED WORK

Existing studies on the interoperability of context-aware systems are limited to a specific domain and do not consider interoperability between different domains. In addition, as far as we know, ontology services have yet to be provided as a separate semantic service in the field of interoperability between context-aware systems. The following describes related studies on providing ontology or service for context-aware systems.

Cabrera et al. [10] presented an architecture for monitoring context-aware systems. This architecture includes a set of monitoring tools, a monitoring orchestrator, an ontology manager, and a repository. For semantic interoperability, the ontology manager performs context modeling, and the repository stores the output of monitoring tools with the structure received from the ontology manager. The ontology presented in this research [11] is a three-level ontology for all IoT environments with context-aware capabilities, and semantic mapping between different domains is not considered.

Hassani et al. [12] proposed a framework for a context management system in IoT. In this research, context is considered as a service. The proposed framework includes a context broker, which is the middle layer between the context provider and context consumer and provides the possibility of reasoning and context sharing in the cloud environment. Semantic interoperability is not mentioned in this research.

Pradeep et al. [13] proposed a model for providing, organizing, and managing context storage for IoT environments. The proposed context model has considered the formal concepts related to objects, including their classification and relationships, the different features, and the challenges of the context-aware ecosystem. Overall, an ontology-based context organization with a three-level hierarchical approach was proposed for a context-aware IoT ecosystem. In this research, semantic mapping between different domains is not considered.

Moradi et al. [14] presented the CaaSSET framework for the model-driven development of complex context services. Their framework is equipped with a comprehensive metamodel for modelling and developing context services. Their metamodel supports three key context concepts: entity, context source, and context. The context is divided into two categories: high-level context and context element, and the high-level context also includes three categories: atomic contexts, aggregated contexts, and derived contexts. CaaSSET facilitates the development of context as a service; however, it has yet to consider semantic interoperability.

De Matos [15] presented an architecture for context sharing based on edge-fog. The edge layer is the context providing system, and the fog layer is the context sharing system. Context information for the interoperability between edge and fog layers is modeled in JSON or XML format. In the edge layer, context information modelling is done ontology, and all context information is stored in the cloud. In this research, context sharing and security issues are among the main challenges, and all semantic interoperability aspects still need to be investigated.

Flahive et al. [16] proposed ontology as a service in the cloud to extract sub-ontology from a more extensive ontology and replace it with a sub-set of a local ontology.

III. MOTIVATIONAL EXAMPLE

This section describes an example of a smart city with different domains, such as a smart home, a smart hospital, and a traffic management system. This example, illustrated in figure 1, is a partial scenario. However, it can demonstrate the concept of context sharing and semantic interoperability requirements in context-aware systems along with the role of broker and ontology service providers.

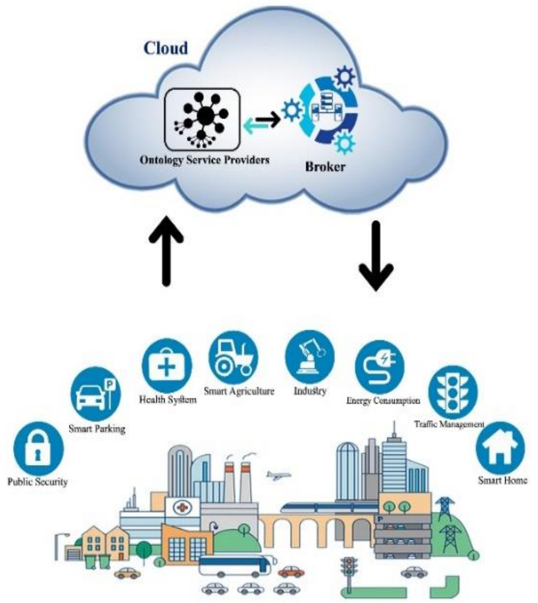

Figure 1 - Context sharing in a smart city scenario

In a smart home scenario where an older person lives alone in a house, there are body temperature, pressure, heart rate, smoke, motion, and other sensors in the smart home to take care of the habitant. By analyzing heart rate sensor data, the context-aware system recognizes that the older person is not in good condition and should report the situation to the health system's family doctor and the emergency department. The smart home system uses the technique to model the context and the rules technique for reasoning and interpreting the context information. However, the ontology technique is used for modelling and reasoning in the health system. The smart home system expresses the temperature in Celsius, but in the health system, it is expressed in Fahrenheit. The temperature feature is entitled “temp” in the smart home system, whereas it is entitled “temperature” in the health system.

After receiving the context information from the smart home system, the ontology service broker sends it to an ontology service provider for semantic analysis. The ontology service provider checks the semantic differences between the two systems and returns the result to the ontology service broker. In the end, the context information is sent to the health system as an ontology service consumer. The ontology service provider is elaborated on in the next section.

IV. OVERVIEW OF THE APPROACH

This section presents the proposed approach to facilitate semantic interoperability of context-aware systems in two parts. In part A, the architecture of OaaS is explained as a method to provide ontology service. In part B, the proposed context ontology is elaborated.

A. OaaS

In this section, we describe the concepts related to OaaS. Figure 2 shows an overview of the OaaS approach. OaaS aims to provide an ontology service to achieve semantic interoperability for ontology service consumers. In the following, we introduce the constituent elements of the proposed architecture:

- Ontology Repository: Each repository contains ontologies related to a specific domain. There is at least one ontology per domain in this repository.
- Ontology Service: It is a web service that explores several domain-specific ontologies to find concepts in the ontologies and map different concepts together.
- Ontology Service Provider: It hosts, manages, and provides ontology and semantic services for the consumer of the ontology service.
- Ontology Service Consumer: It refers to any application, software component, or web service that reuses or consumes provided ontology services to resolve the semantic interoperability challenges.
- Ontology Service Broker: It is an intermediate layer for data exchange between ontology service consumers and ontology service providers. The ontology service broker can to process complex requests.

Ontology service providers consist of several ontologies and metadata repositories, services related to each ontology repository, and service interfaces. Each ontology repository belongs to a domain, and several services are provided for each repository. A service interface is provided to access the ontology services. Ontology service providers provide three primary ontology services::

- Positioning: It receives an element (context concept), searches it, and returns the position of an element in an ontology repository.
- Ontology Schema: This service returns all or part of the ontology.
- Mapping: It is the most the essential service provided by ontology service providers. In the software engineering literature, this service is well-known as ontology alignment. The mapping service investigates the source and destination ontologies in terms of similar names, attributes, and data structure to send data from one domain to another. This service can use the composite of other services to complete its tasks.

The manager/developer of the system designs and deploys the ontology and metadata needed for each business domain. The developer also specifies the required mapping between different domain ontologies. After publishing this information, ontology service consumers or brokers can consume a specific ontology service. In the proposed architecture, the ontology service consumer has the most negligible dependence on the ontology service broker and can send their requests directly to ontology service providers.

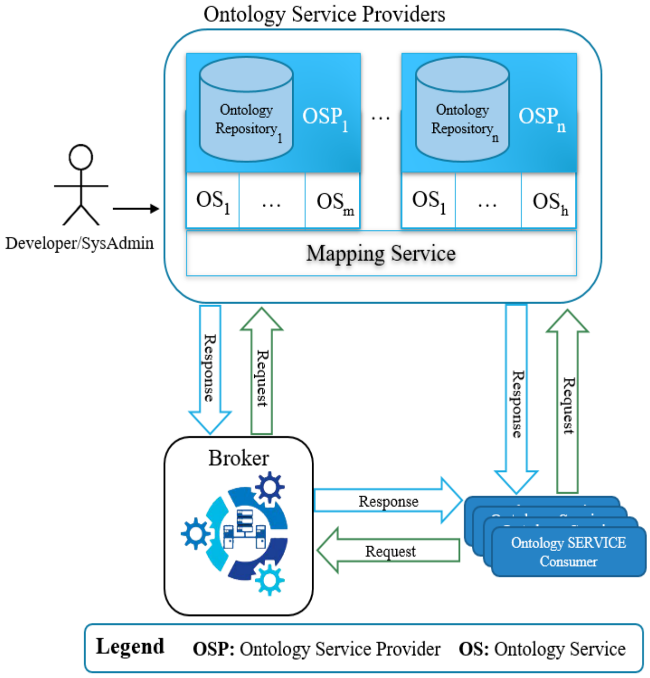

Figure 2. The big picture of the Ontology as a Service approch

B. Proposed ontology

The context sharing process is more complicated than the data sharing process [17]. As explained in section III, the contexts in each domain are modeled and reasoned differently, and semantic interoperability should be considered for sharing contexts between domains. Ontology service providers can act as a solution to facilitate semantic interoperability between context-aware systems. However, we need to define an ontology with a similar structure for each system so that the ontology can be aligned. An ontology with many concepts and relations slows processing and searching [7] and is unsuitable for health systems requiring real-time responses. According to Figure 3, a lightweight ontology is proposed. This ontology is adapted from Cabrera et al. [11] and Pradeep et al. [13]. Note that we customized the ontology and reduced its concepts to reach a lightweight ontology that facilitates semantic interoperability of context-aware systems. The essential concepts that should be considered regarding context information are:

- Entity: It refers to a context-aware object about which the context data and information are gathered and processed. Entities are classified into two main categories: agents (e.g., users or organizations) and resources (e.g., devices, sensors, or software) [11].
- Context Category: Each context data or information are associated with a specific context category.

- Context Source: Context data are originated from different context sources such as sensors, web services, middleware software, or repositories [14]. Note that from the ontological and semantic viewpoint, context sources can be considered a sub-type of resource entities (context-aware objects).

In the example of the older person with a high heart rate, the entity is the older adult, the context specifies the status of the older person, and the context information is obtained from the heart rate sensor. Therefore, in this ontology, there is a need to categorize contexts. Cabrera et al. [14] classified context information into seven categories. “Time” refers to the temporal context. “Profile” is related to the characteristics of the entities. “Environment” is related to the environmental and social conditions that affect an entity. “Role” refers to the role of an entity. “States and Status” is related to the status of an entity at a specific time. “Location” refers to the geographic position of an entity, and “Activity” refers to the set of activities of an entity.

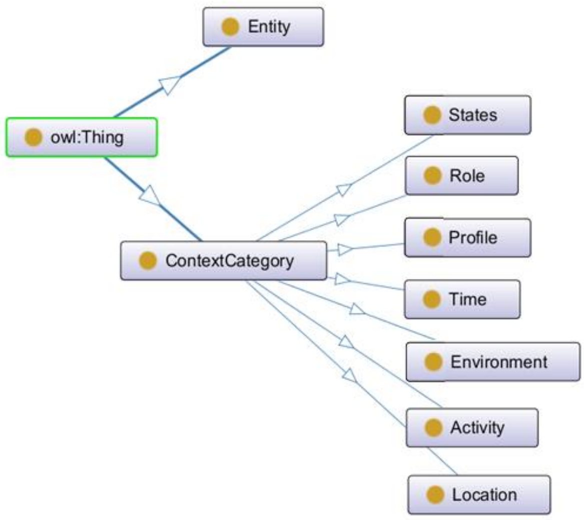

Figure 3- Proposed ontology
(based on Cabrera et al. [11] and Pradeep et al. [13])

The system developer or system administrator uploads the ontology and metadata related to each context-aware system in the repository of the ontology service provider.

V. CASE STUDY

The scenario of the smart home and health systems was considered to evaluate the proposed approach. As illustrated in Figure 4, in the smart home scenario, the context information related to an older person's condition is sent to the ontology service broker to inform the doctor in the health system. Due to the semantic differences between the two systems, the ontology service broker uses OaaS to resolve the semantic interoperability challenges between exchanged contextual data and information.

Note that we considered the three semantic differences mentioned in the introduction, including differences in modeling approach, attribute name, and attribute content (values of attributes).

The proposed ontology creates a uniform model for all systems. The manager/developer of the system configures the ontology alignment and the mapping between domain-specific ontologies. For the mapping process, it is necessary to run the positioning service in each of the repositories to find the exact location of attributes and values.

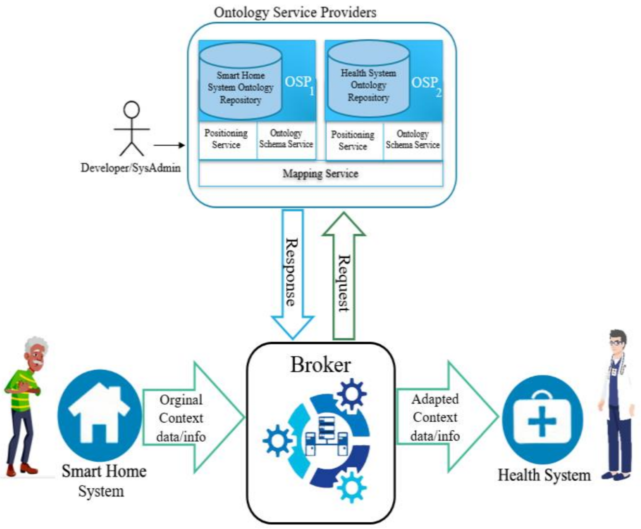

Figure 4- Interoperability of smart home and health system using OaaS

Figure 5 is a simplified ontology with limited concepts of a smart home to check the person's status at home. The "Person" class refers to the name of the person. "Status" subclasses include sensors that control a person's condition, such as "Temp," which refers to body temperature, "Pressure," which refers to a person's pressure, and "HeartRate," which refers to heart rate. The measurement scale and the sensors' data types are specified. This information will help the transformation unit to convert formats to each other when transferring to another system. "Environment" subclasses include sensors controlling the home environment's conditions.

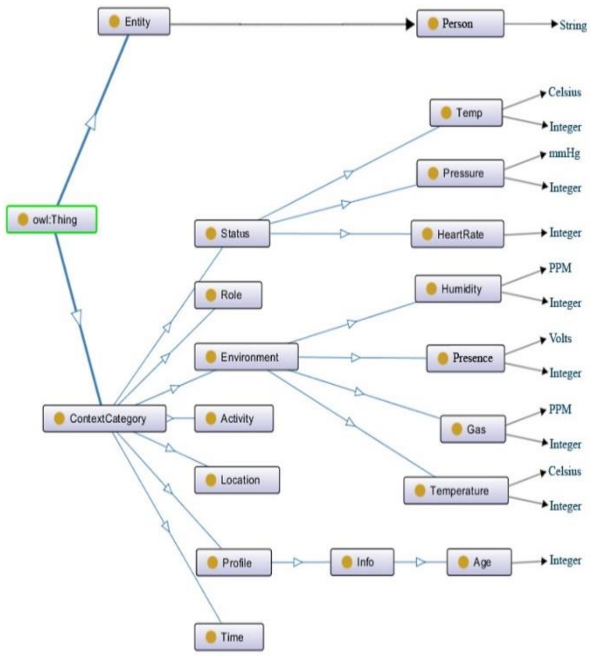

Figure 5- Simplified ontology of smart home
(based on Cabrera et al. [11] and Pradeep et al. [13])

Figure 6 shows the simplified ontology of the health system for receiving patient information. We intend to take the patient's temperature, pressure, and heart rate to check his condition. This ontology includes three semantic differences with the smart home system. Here, the word temperature is displayed with "temperature" and is different from smart homes in which the temperature is displayed with "temp". Moreover, in the smart home system, the temperature is displayed in Celsius, and in the health system, it is displayed in Fahrenheit. Another difference is in the data type of the pressure sensor, which is used in Integer in the smart home system, and String in the health system.

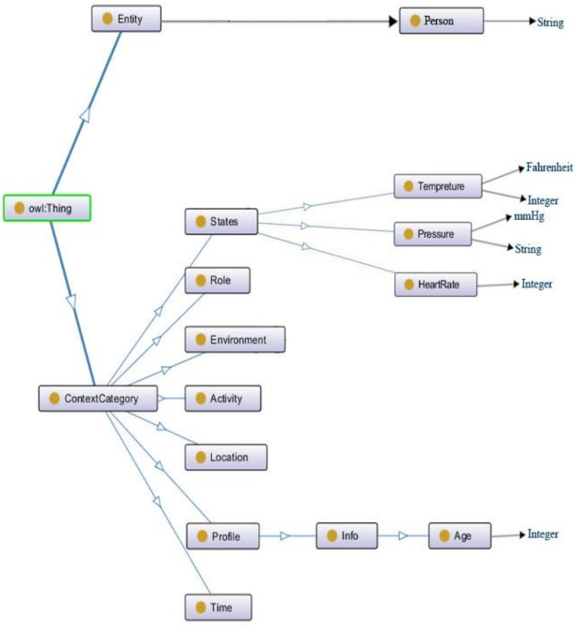

Figure 6- Simplified ontology of health system
(based on Cabrera et al. [11] and Pradeep et al. [13])

After receiving the context information from the smart home system, the broker runs the positioning service from the ontology service provider related to the smart home to determine the element's location in the context category. For example, the word "Temp" is located in the context category related to the ontology under the "Status" class. Then, in the health system, the "status" category finds the word "Temperature" by calling the mapping service and converts Celsius to Fahrenheit by calling the transformation unit. In the same way, in the "Pressure" subclass, the Integer type is converted to the string type. Therefore three types of semantic differences were covered by OaaS.

VI. CONCLUSION AND FUTURE WORK

This paper proposed OaaS architecture as a solution to achieve semantic interoperability between different systems. OaaS provides semantic interoperability requirements as a standalone service, so ontology service consumers can use OaaS by calling the required services. OaaS includes some ontology service providers, each consisting of an ontology repository and some ontology services. Positioning, mapping, and ontology schema are the three primary services provided by an ontology service provider. We showed that OaaS facilitates semantic interoperability of context-aware systems. For this purpose, a lightweight ontology for context-aware systems was proposed. By examining the semantic interoperability for the case study, OaaS was able to provide semantic interoperability between the smart home system and the health system despite the three semantic differences.

Context-aware systems are implemented based on various standards, tools, and architectures. In the future, we intend to work on applying OaaS in real-world industrial projects. Moreover, context reasoning is critical for extracting high-level context from underlying contextual data. Hence, in future work, we aim to work on the design and develop a context reasoning service based on the proposed OaaS approach.

REFERENCES

[1] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, "Towards a Better Understanding of Context and Context-Awareness," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 1707, pp. 304–307, 1999, doi: 10.1007/3-540-48157-5_29.

[2] R. C. Motta, K. M. de Oliveira, and G. H. Travassos, "A conceptual perspective on interoperability in context-aware software systems," *Inf. Softw. Technol.*, vol. 114, pp. 231–257, Oct. 2019, doi: 10.1016/J.INFSOF.2019.07.001.

[3] ISO/IEC/IEEE, "15939-2017 - ISO/IEC/IEEE International Standard - Systems and software engineering--Measurement process," *IEEE*, vol. 2017, no. IEEE, 2017, doi: 10.1109/IEEESTD.2017.8016712.

[4] R. González-Usach, D. C. Yacchirema-Vargas, M. Jilian-segui, and C. E. Palau Salvador, *Interoperability in IoT. Handbook of Research on Big Data and the IoT*. 2019.

[5] E. S. Pramukantoro, F. Andri Bakhtiar, B. Aji, and R. Pratama, "Middleware for Network Interoperability in IoT," in *2018 5th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, 2018, vol. 2018, pp. 499–502, doi: 10.1109/EECSI.2018.8752917.

[6] O. Cabrera, X. Franch, and J. Marco, "Ontology-based context modeling in service-oriented computing: A systematic mapping," *Data Knowl. Eng.*, vol. 110, pp. 24–53, Jul. 2017, doi: 10.1016/J.DATAK.2017.03.008.

[7] M. Bermudez-Edo, T. Elsaleh, P. Barnaghi, and K. Taylor, "IoT-Lite: a lightweight semantic model for the internet of things and its use with dynamic semantics," *Pers. Ubiquitous Comput.*, vol. 21, no. 3, pp. 475–487, Jun. 2017, doi: 10.1007/s00779-017-1010-8.

[8] T. Elsaleh, M. Bermudez-Edo, S. Enshaeifar, S. T. Acton, R. Rezvani, and P. Barnaghi, "IoT-Stream: A Lightweight Ontology for Internet of Things Data Streams," in *2019 Global IoT Summit (GIoTS)*, Jun. 2019, pp. 1–6, doi: 10.1109/GIOTS.2019.8766367.

[9] M. Bermudez-Edo and K. Taylor, "IoT-Lite Ontology," 2015, Accessed: Nov. 04, 2022. [Online]. Available: http://www.w3.org/Submission/2015/SUBM-iot-lite-20151126/Latestversion:http://www.w3.org/Submission/iot-lite/.

[10] O. Cabrera, M. Oriol, X. Franch, and J. Marco, "A context-aware monitoring architecture for supporting system adaptation and reconfiguration," *Computing*, vol. 103, no. 8, pp. 1621–1655, Aug. 2021, doi: 10.1007/s00607-021-00923-z.

[11] O. Cabrera, X. Franch, and J. Marco, "3LConOnt: a three-level ontology for context modelling in context-aware computing," *Softw. Syst. Model.*, vol. 18, no. 2, pp. 1345–1378, Apr. 2019, doi: 10.1007/s10270-017-0611-z.

[12] A. Hassani *et al.*, "Context-as-a-Service Platform: Exchange and Share Context in an IoT Ecosystem," in *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, Mar. 2018, pp. 385–390, doi: 10.1109/PERCOMW.2018.8480240.

[13] P. Pradeep, S. Krishnamoorthy, R. K. Pathinarupothi, and A. V. Vasilakos, "Leveraging context-awareness for Internet of Things ecosystem: Representation, organization, and management of context," *Comput. Commun.*, vol. 177, pp. 33–50, Sep. 2021, doi: 10.1016/J.COMCOM.2021.06.004.

[14] H. Moradi, B. Zamani, and K. Zamanifar, "CaaSSET: A Framework for Model-Driven Development of Context as a Service," *Futur. Gener. Comput. Syst.*, vol. 105, pp. 61–95, Apr. 2020, doi: 10.1016/j.future.2019.11.028.

[15] É. de Matos, "Edge-centric context sharing architecture for the internet of things : context interoperability and context-aware security," Pontifícia Universidade Católica do Rio Grande do Sul, 2020.

[16] A. Flahive, D. Taniar, and W. Rahayu, "Ontology as a Service (OaaS): extracting and replacing sub-ontologies on the cloud," *Clust. Comput.*, vol. 16, no. 4, pp. 947–960, Oct. 2012, doi: 10.1007/S10586-012-0231-X.

[17] E. de Matos *et al.*, "Context information sharing for the Internet of Things: A survey," *Comput. Networks*, vol. 166, p. 106988, Jan. 2020, doi: 10.1016/J.COMNET.2019.106988.