Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 56-2 (2023) 9831–9836

Interoperability Middleware for IIoT Gateways based on Standard Ontologies and AAS

Pedro Henrique Morgan Pereira * **Gustavo Cainelli** **
Carlos Eduardo Pereira *** **Edison Pignaton de Freitas** ****

* *SENAI Institute of Innovation in Integrated Solutions in Metal Mechanics, São Leopoldo, Brazil, (e-mail: pedro.pereira@senairs.com.br).*
** *Institut für Automation und Kommunikation e.V. Magdeburg (ifak), Germany (e-mail: gustavo.cainelli@ifak.eu)*
*** *Graduate Program of Electrical Engineering (PPGEE), Universidade Federal do Rio Grande do Sul, Porto Alegre, Brazil, (e-mail: cpereira@ufrgs.br)*
**** *Graduate Program of Electrical Engineering (PPGEE), Universidade Federal do Rio Grande do Sul, Porto Alegre, Brazil, (e-mail: edison.pignaton@ufrgs.br)*

Abstract: Recent advancements in microelectronics, information technology, and communication protocols have led to smaller devices with greater processing capabilities and lower energy consumption. These developments have contributed to the growing number of physical devices used in industrial environments that are interconnected and communicate through the internet, allowing for the realization of Industry 4.0 and the Industrial Internet of Things (IIoT). Numerous companies are involved in producing these devices, each using different communication protocols, data structures, and IoT platforms. This has resulted in interoperability issues that need to be addressed. To mitigate these issues, this paper proposes an interoperability middleware for IIoT gateways that adopts ontologies based on international standards (IEEE 1872-2015) and the asset administration shell (AAS) using industrial frameworks.

Keywords: Interoperability, Industry 4.0, IIoT, Middleware, Gateway, Asset Administration Shell, Information Integration, Industrial Ontology.

1. INTRODUCTION

Interoperability in IoT refers to the ability of two systems to communicate and share services (Noura et al. (2019)). With the increasing number of physical devices used in industrial environments that are interconnected and communicate via the internet, concepts such as Industry 4.0, Industrial Internet of Things, and Cyber-Physical Systems have become possible (Qin et al. (2016); Younan et al. (2020)). However, due to the heterogeneity of devices developed by different companies and the use of diverse communication protocols, data structures, and IoT platforms, interoperability issues arise. According to the European project Unify IoT (Aazam et al. (2018)), over 300 IoT platforms are developed annually in the current market.

There are at least five distinct interoperability perspectives (Noura et al. (2019)), identified as i) device interoperability, ii) syntactical interoperability, iii) semantic interoperability, iv) network interoperability, and v) platform interoperability. Each of these perspectives addresses interoperability issues at a different level, but they are all equally important to ensure the development of an interoperable system.

Semantic technologies play a critical role in addressing interoperability challenges in Industry 4.0. By using ontologies to establish a common understanding of data and relationships between system elements, these technologies enable effective communication and interaction between machines and humans, unlocking the full potential of Industry 4.0. (Mayer et al. (2017)).

To ensure that different assets and platforms can work together seamlessly, it is not enough to simply exchange process data, but also manage and control data for system elements. To address this challenge, this work proposes a middleware for IIoT gateways (Papcun et al. (2020); Aazam et al. (2018)) to mitigate interoperability problems. The proposed approach adopts international standard ontologies combined with assets standardized digital representation as the Asset Administration Shell (AAS). Specifically, the middleware focuses on three of the interoperability perspectives (Noura et al. (2019)): device, syntactical, and semantic interoperability.

The two main contributions of this work are:

- a general Industry 4.0/IIoT oriented ontology based on international standards;

2405-8963 Copyright © 2023 The Authors. This is an open access article under the CC BY-NC-ND license.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2023.10.403

- a middleware implementation that integrates the developed ontology with the AAS concept, to mitigate interoperability problems in industrial environments.

The remainder of this article is organized as follows. Section 2 gives necessary background information on ontology and asset administration shell. Section 3 presents related works that mitigate interoperability problems in industrial environments using ontologies and AAS. Section 4 explicitly describes the hardware used for the IIoT gateway, the middleware's architecture, introduces both the general and the AAS use case and presents the developed ontology. Section 5 demonstrates the performed simulations and its results. Finally, Section 6 gives the summary and the outlook.

2. BACKGROUND

This section gives a brief overview of ontologies and AAS which are key elements of the proposed approach.

2.1 Ontologies

IIoT applications, such as smart manufacturing or intelligent maintenance systems, often encounter interoperability issues, particularly related to semantic and syntactic differences in exchanged data. One way to address these problems is to use ontologies, which provide a shared vocabulary and a structured representation of domain concepts and relationships that machines can interpret (Wang et al. (2004)).

To specify ontology concepts, it is necessary to define key properties, relationships, and axioms for the given domain. Removing ambiguities and establishing a common understanding of the system among different nodes (Bajaj et al. (2017)). Thus, using ontologies can be a powerful solution to improve interoperability in IIoT applications, since it enables unambiguous structured data sharing among nodes and performs reasoning over it (de Freitas et al. (2020)).

2.2 AAS

The new paradigm of industry 4.0 is the digitalization of assets. The key elements for that are standardization and interoperability. The Platform Industrie 4.0 (2022) proposes the Asset Administration Shell (AAS) which is a standardized digital representation of industrial assets based on the Architecture Model for I4.0 (RAMI 4.0) (Hankel and Rexroth (2015)). It enables the implementation of Digital Twins and interoperability between automated industrial systems and Cyber-Physical Systems, which means the AAS is the core element for an interoperable description of assets. The AAS is supposed to have a relevant role for further developments in the Industry 4.0 landscape. However, due to the complexity of the standardization ecosystem, there are still few scenarios implemented.

Assets are valuable elements of an organization. Examples are devices, machines, components, documents, or even software. The AAS provides semantical meta-data of a CPS including functional and non-functional properties (Nagorny et al., 2018) and digital models of various aspects, playing a pivotal role in establishing communication among I4.0 components and managing interoperability between the applications and the manufacturing systems (Platform Industrie 4.0, 2022).

Each AAS has a header and a body. The header contains a list of parameters that identify the AAS and the physical asset. The body stores data related to the capabilities of an asset and its operational data. The asset's information is described as submodels in the AAS. The AAS may incorporate general submodels (e.g. identification) and also specific submodels, e.g. communication (Ye and Hong, 2019). The submodels contain submodel elements like SubmodelCollection and Properties. A property, which is a submodel element type, can contain a value that represents a physical variable of the asset. It can be of several types as INT, BOOL, or STRING. An example is the current status of a pump (ON / OFF).

3. RELATED WORK

With the increasing dissemination of applications based on IoT nodes, interoperability becomes a major challenge and several approaches have been published in the literature to handle this problem.

The author of Steinmetz et al. (2018) presents how to map industrial elements into a semantic model, integrating Industry 4.0 objects easily and understandably for the users. These semantic models can support services, and allow data exchange between virtual and physical assets through an IoT middleware.

The used semantic model is an I4.0 ontology, which is composed of other ontologies found in the literature that bring complementary concepts for IoT applications. As an example, the IoT Lite ontology (Bermudez-Edo et al. (2016)) is an instance of Semantic Sensor Network (SSN) ontology (Compton et al. (2012)), which describes key concepts of IoT, such as devices, sensors, attributes, and so on. Nevertheless, the author added new classes related to the type of information visualization that allows the modeling of how data should be presented to different types of users. The model was used to generate interfaces to the FIWARE IoT middleware and validated through an industrial valve actuator use case. The valve circuit was instrumented with different kinds of sensors to monitor real-time data, such as battery level, temperature, pressure, and torque. Although the author has carried out experiments with different types of sensors, they were not treated as different devices. Therefore, the developed ontology does not handle problems related to communication between different devices with heterogeneous communication protocols, such as the one presented in this work.

The authors of Iñigo et al. (2020) present a case study on the application of the AAS in an industrial context. The use case considers a plant composed of a robotic arm, a grinding machine, and a semantic harmonization layer. It was created two AASs (Robotic Arm AAS and Grinding Machine AAS), and both were tested using a semantic integrator for experimentally validated interoperability. The demonstrator has been transformed to consider the RoboticArm as an Asset and it used a Raspberry Pi as the OPCUA server that publishes the Administration Shell of the RoboticArm asset instance. The developed architecture provided a protocol translator that exchanges information using UMATI (Universal Machine Tool Interface), an interface that standardizes the way the machine tools share information over OPCUA. In conclusion, the AAS has been validated, not only as a representation of heterogeneous industrial assets and their digital twins but also to enable the interoperability between them in a manufacturing plant.

The interoperability experiments carried out and reported in Iñigo et al. (2020) were composed of two IoT devices, both communicating via OPC-UA communication protocol. The AAS use case for this work has three IoT devices, each communicating with different communication protocols and data structures. As the authors supposed, to implement a larger AAS scenario the use of a methodology, such as the I4.0 ontology developed in this work, is of paramount importance to mitigate interoperability problems concerning the AAS.

4. METHODOLOGY

This section presents the steps and design choices taken to develop the proposed Industrial Interoperability Middleware.

4.1 Architecture

The middleware's architecture is composed of three main blocks: Data Storage, Communication, and User Interface.

The "Data Storage" block is responsible for storing physical and digital data of industrial assets, also known as digital twins, in either a local or cloud database. Before being saved, all incoming messages are processed. The ontology description enables the identification of various information from the message, such as the device ID that sent the message, the communication protocol used, and the type of asset (sensor or actuator).

The database stores structured data, including information for each sensor and actuator individually. Each message is time-stamped and has different labels defining the sender's communication protocol, asset type, and other relevant details. This makes it easier to read and write both processed and incoming data. It's worth noting that the saved data are related to industrial processes such as production lines, maintenance systems, etc. Thus, the stored data can serve as a dataset for machine learning algorithms, enabling the prediction of machine failures and the improvement of manufacturing time.

The "User Interface" block includes a Supervisory Control And Data Acquisition (SCADA) developed using the programming tool Node-RED (O'Leary and Conway-Jones (2016)). The developed SCADA-like system is used for industrial process monitoring, as well as for configuring the communication parameters of brokers and servers. Users can intervene manually in the process.

The last block, "Communication", allows data transmission to/from different industrial assets by incorporating different communication protocol servers and brokers. This block can handle data from real or digital versions of industrial assets. This work has depicted an application using three communication protocols that are widely used in the industry: OPC UA, MQTTP, and DDS. OPC UA is adopted as an intermediary protocol to which the others are translated. Two translators for publish-subscribe communication protocols to OPC UA, MOB (MQTT OPC UA Bridge), and DOB (DDS OPC UA Bridge) were developed. The payload received by the gateway from MQTT or DDS messages will be translated to OPC UA and stored in the server. The bridging mechanisms run as ROS2 (ROS2 (2021)) nodes in a dedicated workspace, written in python, and are dependent on the communication brokers and the IIoT ontology.

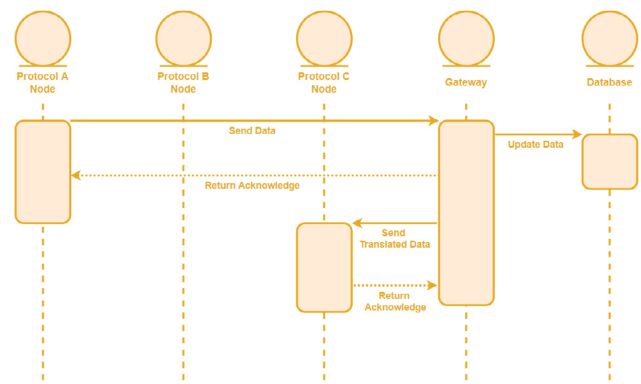

Fig. 1. Sequence UML diagram for data exchange between devices.

Figure 1 depicts a UML sequence diagram illustrating the exchange of messages between three devices, each utilizing a different communication protocol (A, B, and C). Node C requires data that is transmitted by Node A. In this scenario, Node A sends a message to the gateway, which saves the received data to the database and sends back an acknowledgment message. The middleware of the gateway then translates the message from protocol A to the intermediate protocol B, and subsequently translates the data from protocol B to C. The translated data is then transmitted to Node C, which sends an acknowledgment message back to the gateway.

4.2 Hardware

The interoperability middleware was developed for a gateway based on a Raspberry Pi 4 with 8 GB of RAM, and 128 GB NVMe memory. This gateway runs a Ubuntu Server 20.04 LTS and includes a bridging mechanism for industrial communication protocols such as MQTT, DDS, and OPC UA. In addition, the gateway is equipped with data pre-processing capabilities such as data filtering and data consolidation, as well as local data storage using the OPC UA server and InfluxDB.

4.3 IIoT Interoperable Ontology

The proposed ontology, IIoTOnTo, is based on the IEEE Standard Ontologies for Robotics and Automation (IEEE-SA (2015)), and was developed with Protégé (Musen (2015)), an open-source tool for creating and editing domain ontologies, using the web ontology language (OWL) (Group et al. (2009)), which is well-established

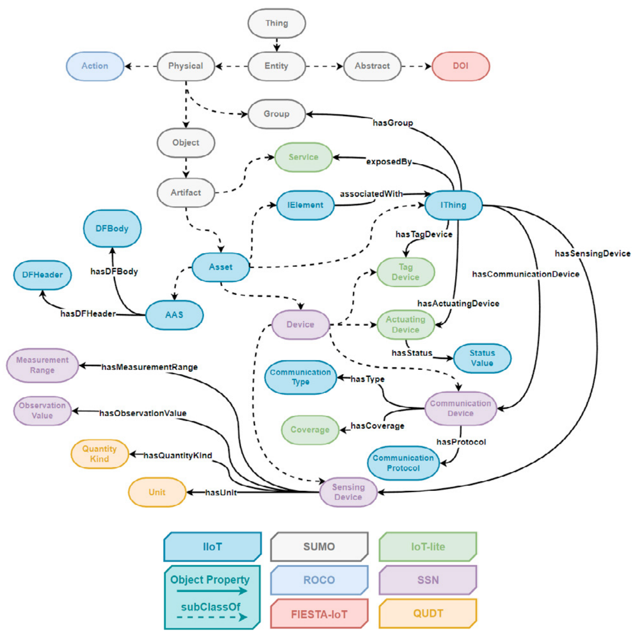

Fig. 2. Part of the proposed ontology, IIoTOnTo.

for knowledge-based applications. The top-level ontology uses categories defined in the SUMO ontology and adapts classes from CORA, CORAX, and POS axioms (de Freitas et al. (2020)). These ontologies define essential concepts of robot and automation applications, such as behavior, function, goal, and task, which can be extended to this work applications by incorporating IIoT-specific ontological concepts. Nevertheless, components from ontologies developed by W3C Semantic Sensor Network Incubator Group were also included in the interoperability ontology, such as QUDT (Rijgersberg et al. (2013)), SSN (Compton et al. (2012)), and IoT-Lite (Bermudez-Edo et al. (2016)) Ontologies. Fig. 2 represents the combination of classes from different base ontologies, identified by different colors, and the developed classes for this proposal, entitled IIoT. However, the image presents only a part of the ontology since it comprises more than 450 classes.

In addition to dealing with semantic interoperability, the ontology description is the base of two middleware configuration files. One of these files, "config.yaml", defines the configurations for the local database and for the communication protocols brokers/servers, such as the broker's URL, port, and credentials for MQTT. The other file describes the IoT devices from their communication protocol to their sensors, actuators, and data dependencies. Therefore, whenever a new asset is included in the application, the gateway must generate both configuration files to reconfigure the communication protocol translator scripts.

4.4 Use case

The selected use case to validate the proposed middleware it is an autonomous production of a nutrient solution intended for agricultural purposes. Soilless cultivation is a method for growing plants without soil but using either inert organic or inorganic growing substrates combined with a nutrient solution (Nerlich and Dannehl (2021)). Compared to standard techniques, soilless culture (hydroponics and aeroponics techniques) is more attractive since it can

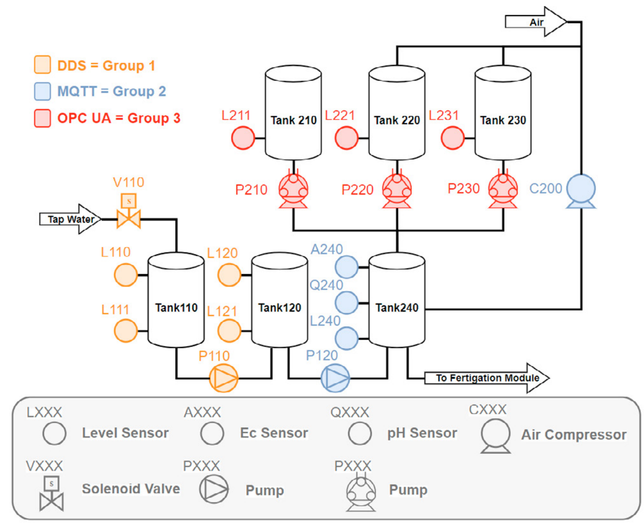

Fig. 3. IIoT Use case.

control water availability, pH, and nutrient concentrations (Papadopoulos et al. (2008)). The autonomous production of this solution demands real-time environmental monitoring and remote control of a sequence of processes. These needs are met by using heterogeneous devices equipped with different sensors and actuators. These devices can use distinct communication protocols and data structuring, which increases interoperability problems.

The use case is represented by Fig. 3, which is composed of seven actuators (one valve, five pumps, and one air compressor) and ten sensors (eight levels, one pH, and one Ec sensor), which were divided in three groups, each controlled by an IoT device. Each the group is identified by a color and communicates by a different communication protocol. The group identified by the orange color is monitored/controlled by a DDS node, the blue one by an MQTT node, and at last, the red one by an OPC UA Client node.

4.5 AAS Use Case

To address semantic interoperability issues, a virtual representation of all assets used in the case study was created. The representation involved developing twenty-one Asset Administration Shells (AAS), which included seven actuators, ten sensors, three devices, and one gateway. Each AAS identified the critical characteristics of its corresponding physical asset. Figure 4 displays the assets in Group 1 and their respective AAS, which consisted of a DDS node (D001), four level sensors (L110, L111, L120, and L121), a pump (P110), and a solenoid valve (V110).

5. SIMULATION AND RESULTS

The interoperability experiments were conducted based on the industrial application presented in subsection 4.4, in which three devices communicate via different IoT protocols. Python scripts were used to create digital twins of the assets through their AAS for the experiments.

Initially, the assets are described, including gateways and IoT nodes such as sensors and actuators, based on the

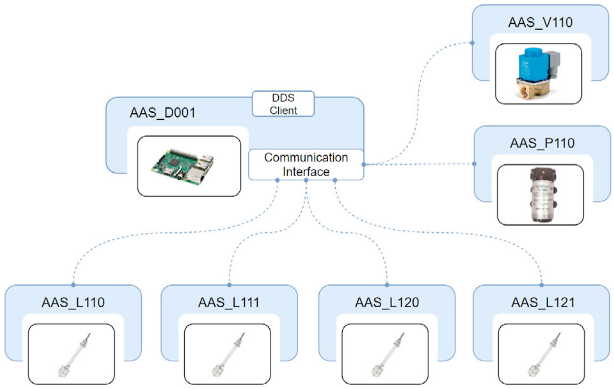

Fig. 4. IIoT Interoperability AAS use case.

developed ontology. These descriptions comprise the information required to generate two gateway configuration files related to the communication protocol translators. Moreover, it serves as the basis for the development of AAS, which is a crucial concept used to mitigate semantic interoperability issues.

Once the gateway is configured, the user can initiate the experiments by running the OPC-UA server and the communication bridge scripts through the SCADA-like system. Once the translators are active, the digital twin scripts of the three devices can be initialized.

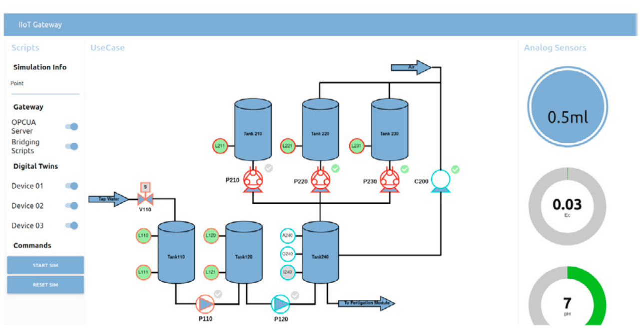

Fig. 5. Screenshot of the SCADA-like system, while running the experiments.

During the experiment, three scripts are used to emulate three real assets, representing the active part of the virtual representation of each asset, which is its digital twin. The communication bridges translate the process data of the three devices, which is then stored in the OPC UA server. The SCADA-like system collects the devices' data from the OPC UA server, enabling the user to monitor the devices' current status (Fig. 5). All experimental data is stored locally in the designated InfluxDB point for future analysis and can be used for creating datasets for machine learning algorithms.

The results of the use case simulation are presented in Fig. 6, by storing data from all sensors and actuators using any of the select communication protocols(MQTT, DDS, and OPC-UA). By analyzing the data, it is possible to observe seven cycles of nutrient solution production during the experiment. This is demonstrated by the seven plateaus, each representing the solution's pH value of 7 (the pH value of water) in "Tank 240" at the beginning of each cycle. Additionally, at least one production cycle was successfully completed, demonstrating the establishment of communication between the devices.

Since InfluxDB works with tags, it is possible to filter data by the device's communication protocol to the asset's id and type (sensor or actuator). As an example, it is possible to filter data from only actuators communicating by MQTT protocol. It is worth mentioning that all data sent by MQTT or DDS translates to OPC-UA; therefore, it is helpful to use this communication protocol filter to check for application errors.

6. CONCLUSION

The purpose of this paper is to introduce a middleware solution for IIoT gateways that aims to address interoperability issues in industrial environments. The approach taken in this project combines ontologies based on international standards (IEEE), communication protocol translators, and AAS. This solution is designed to mitigate device, syntactical, and semantic interoperability problems that commonly arise in IIoT applications.

The validation of the middleware was carried out in an industrial use case that involved the production of nutrient solutions for agriculture using three IoT devices. The middleware successfully enabled communication between heterogeneous assets with different communication protocols and data structures. Additionally, it digitized the most relevant information about the assets through AAS, creating digital twins of the assets. Furthermore, the middleware allowed users to control the simulation and monitor real-time data using a SCADA-like system. The simulation results showed strong evidence that the developed ontology combined with AAS is useful for various industrial applications involving IoT devices. Although the ontology was designed for general industrial applications, its effectiveness was evaluated in a specific use case.

In future work, our group intends to conduct a simulation using physical devices to compare the results obtained in the digital world with those in the physical world. Additionally, we plan to extend the implementation of the AAS and validate it in a different scenario using the same ontology. Finally, our goal is to develop a parser that can autonomously create AAS from ontology files.

REFERENCES

Aazam, M., Zeadally, S., and Harras, K.A. (2018). Deploying fog computing in industrial internet of things and industry 4.0. *IEEE Transactions on Industrial Informatics*, 14(10), 4674–4682.

Bajaj, G., Agarwal, R., Singh, P., Georgantas, N., and Issarny, V. (2017). A study of existing ontologies in the iot-domain. *arXiv preprint arXiv:1707.00112*.

Bermudez-Edo, M., Elsaleh, T., Barnaghi, P., and Taylor, K. (2016). Iot-lite: a lightweight semantic model for the internet of things. In *2016 INTL IEEE conferences on ubiquitous intelligence & computing, advanced and trusted computing, scalable computing and communications, cloud and big data computing, internet of people, and smart world congress (uic/atc/scalcom/cbdcom/iop/smartworld)*, 90–97. IEEE.

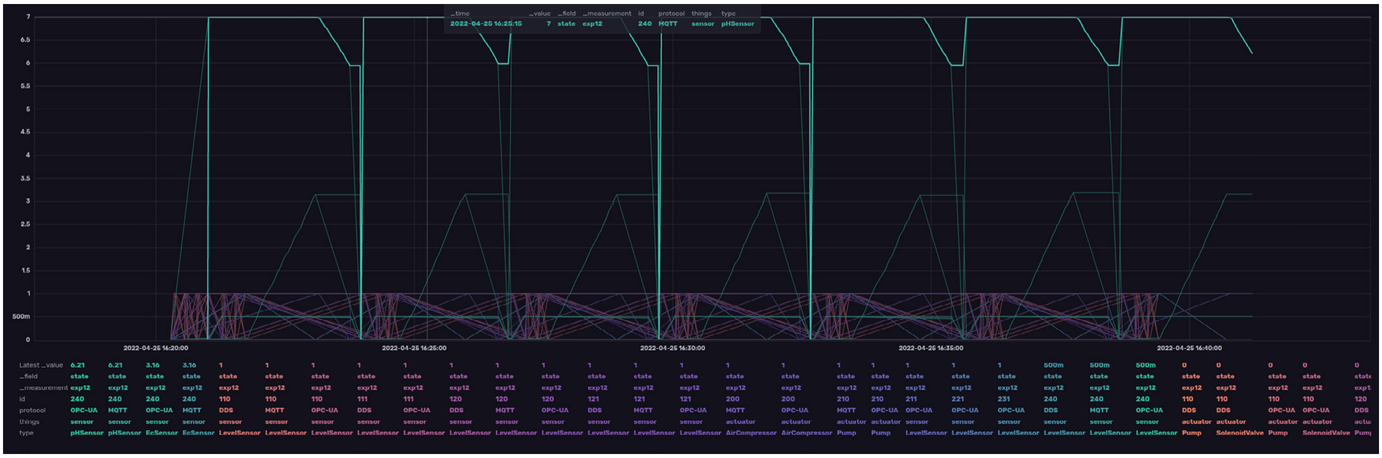

Fig. 6. InfluxDB simulation data.

Compton, M., Barnaghi, P., Bermudez, L., Garcia-Castro, R., Corcho, O., Cox, S., Graybeal, J., Hauswirth, M., Henson, C., Herzog, A., et al. (2012). The ssn ontology of the w3c semantic sensor network incubator group. *Journal of Web Semantics*, 17, 25–32.

de Freitas, E.P., Bermejo-Alonso, J., Khamis, A., Li, H., and Olszewska, J.I. (2020). Ontologies for cloud robotics. *The Knowledge Engineering Review*, 35.

Group, O.W. et al. (2009). Owl 2 web ontology language document overview: W3c recommendation 27 october 2009.

Hankel, M. and Rexroth, B. (2015). The reference architectural model industrie 4.0 (rami 4.0). *ZVEI*, 2(2), 4–9.

IEEE-SA (2015). IEEE Standard Ontologies for Robotics and automation. URL `https://standards.ieee.org/standard/1872-2015.html`.

Iñigo, M.A., Porto, A., Kremer, B., Perez, A., Larrinaga, F., and Cuenca, J. (2020). Towards an asset administration shell scenario: A use case for interoperability and standardization in industry 4.0. In *NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium*, 1–6. IEEE.

Mayer, S., Hodges, J., Yu, D., Kritzler, M., and Michahelles, F. (2017). An open semantic framework for the industrial internet of things. *IEEE Intelligent Systems*, 32(1), 96–101.

Musen, M.A. (2015). The protégé project: a look back and a look forward. *AI Matters*, 1(4), 4–12. doi:10.1145/2757001.2757003. URL `https://doi.org/10.1145/2757001.2757003`.

Nagorny, K., Scholze, S., Ruhl, M., and Colombo, A.W. (2018). Semantical support for a cps data marketplace to prepare big data analytics in smart manufacturing environments. In *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*, 206–211. IEEE.

Nerlich, A. and Dannehl, D. (2021). Soilless cultivation: dynamically changing chemical properties and physical conditions of organic substrates influence the plant phenotype of lettuce. *Frontiers in Plant Science*, 11, 601455.

Noura, M., Atiquzzaman, M., and Gaedke, M. (2019). Interoperability in internet of things: Taxonomies and open challenges. *Mobile networks and applications*, 24(3), 796–809.

O'Leary, N. and Conway-Jones, D. (2016). Node-red programming tool. Available at: ¡`https://github.com/node-red/node-red`¿. Accessed on: 16 march 2022.

Papadopoulos, A.P., Bar-Tal, A., Silber, A., Saha, U.K., and Raviv, M. (2008). Inorganic and synthetic organic components of soilless culture and potting mixes. *Soilless culture theory and practice. Elsevier, Burlington, MA.*

Papcun, P., Kajati, E., Cupkova, D., Mocnej, J., Miskuf, M., and Zolotova, I. (2020). Edge-enabled iot gateway criteria selection and evaluation. *Concurrency and Computation: Practice and Experience*, 32(13), e5219.

Platform Industrie 4.0 (2022). Details of the asset administration shell - part 1 - the exchange of information between partners in the value chain of industrie 4.0 (version 3.0rc02).

Qin, Y., Sheng, Q.Z., Falkner, N.J., Dustdar, S., Wang, H., and Vasilakos, A.V. (2016). When things matter: A survey on data-centric internet of things. *Journal of Network and Computer Applications*, 64, 137–153.

Rijgersberg, H., Van Assem, M., and Top, J. (2013). Ontology of units of measure and related concepts. *Semantic Web*, 4(1), 3–13.

ROS2, R.O.S.. (2021). The robot operating system galactic geochelone. Available at: ¡`https://docs.ros.org/en/galactic/index.html`¿. Accessed on: 16 march 2022.

Steinmetz, C., Rettberg, A., Ribeiro, F.G.C., Schroeder, G., Soares, M.S., and Pereira, C.E. (2018). Using ontology and standard middleware for integrating iot based in the industry 4.0. *IFAC-PapersOnLine*, 51(10), 169–174.

Wang, X.H., Zhang, D.Q., Gu, T., and Pung, H.K. (2004). Ontology based context modeling and reasoning using owl. In *IEEE annual conference on pervasive computing and communications workshops, 2004.*, 18–22. Ieee.

Ye, X. and Hong, S.H. (2019). Toward industry 4.0 components: Insights into and implementation of asset administration shells. *IEEE Industrial Electronics Magazine*, 13(1), 13–25.

Younan, M., Houssein, E.H., Elhoseny, M., and Ali, A.A. (2020). Challenges and recommended technologies for the industrial internet of things: A comprehensive review. *Measurement*, 151, 107198.