ISSN 2237-8960 (Online)

RESEARCH PAPER

Linked data to streaming data sensors sources for direct access

Diego Calvetti[1], Daniel Luiz de Mattos Nascimento[2], Flávio Ney Magno de Araújo Filho [3], Rafael Henrique Viana Abreu[3], Nicolas Alexandros Papadopoulos[3]

[1]CONSTRUCT/Gequaltec, University of Porto, Porto, Portugal.
[2]University of Barcelona Business School, Barcelona, Spain.
[3]CERTI Foundation, Florianópolis, SC, Brazil.

How to cite: Calvetti, D. et al. (2024), "Linked data to streaming data sensors sources for direct access", *Brazilian Journal of Operations and Production Management*, Vol. 21, No. 1, e20241993. https://doi.org/10.14488/BJOPM.1993.2024

ABSTRACT

Goal: Industrial operations are complex, and data sensors assure safety and reliable information for production improvements. Multiple stakeholders can take advantage of data acquisition for post-analysis and process control. Providing users and systems with friendly access to operating data is fundamental to the digital transition in the industry 4.0 scenario. Linking data and systems over ontologies and Industry Foundation Classes will boost supply chain performance in many layers. This paper presents the concept of valid data points over Uniform Resource Identifiers for sensor time-series into triples stores via Application Programming Interfaces.

Design/Methodology/Approach: A streaming data source approach to integrating industrial sensor data and sharing it via Uniform Resource Identifiers is developed and tested using Node-Red with multiple data connection types, such as the Industry Foundation Classes and open-source time series databases.

Results: The detailed proof of concept presented valid the feasibility of sharing sensor data via Uniform Resource Identifiers. The findings provide a backbone of a system able to interop Message Queuing Telemetry Transport data, Resource Description Framework datasets and Industry Foundation Classes schema.

Limitations of the investigation: The system envisaged was tested using simulated data. However, it is expected to have similar results from real data use. Nevertheless, more research will be needed to implement more features, such as three-dimensional object integration.

Practical implications: The solution designed and tested presented can be used in practice for companies that desire to expand via linked data shareability and interoperability. Also, researchers can advance the solution for specific features, such as creating an open-source data query and manipulation language.

Originality/Value: This paper examines future deployments of systems-to-systems interoperability targeting user-friendly data shareability. It is meant to be useful for industrial and academic developments.

Keywords: Industry 4.0; Linked systems; Server-Sent Events; Time-series; MQTT.

INTRODUCTION

Multiple sensors and systems are vital to industrial operations (Kalsoom *et al.*, 2020; Javaid *et al.*, 2021). Industry 4.0 deployments will lead to increased efficiency (Rehman, 2023). Data and systems interoperability is a primary challenge (Kenda *et al.*, 2019; Kalsoom *et al.*, 2020). Unifying access to numerous data sources in a user-friendly environment is prone to achieve valuable linked data. The Databus platforms concept can enable linked data for users and machines (Dojchinovski *et al.*, 2021; Frey and Hellmann, 2021). Multiple stakeholders over distinguished systems demand information in almost real-time and time-series for post-processing. Data are acquired from plural sensors in diversified systems environments. Based on that, the use of linked data to

Financial support: None.
Conflict of interest: The authors have no conflict of interest to declare.
Corresponding author: diegocalvetti@fe.up.pt
Received: 14 September 2023.
Accepted: 07 December 2023.
Editor: Julio Vieira Neto.

streaming data sensor sources for direct access can be a relevant actor to optimise information transactions and use.

The Application Programming Interfaces (APIs) being developed in the industrial projects for interoperability had many types of databases where Resource Description Framework (RDF) is a vital element (Hernawan *et al.*, 2021; Daquino *et al.*, 2022; Vega-Gorgojo, 2022), but not unique. The industrial supply chain already has many robust and established information systems whose swap/integration is not easy, immediate, and not always desired. In this, systems-to-systems integration is crucial for the sector's digital transition.

The Building Information Modelling (BIM) implementation increases stakeholders' knowledge access, including for the public sector (Schery *et al.*, 2023). The Industry Foundation Classes (IFC) standard data schema and the Linked data ontologies initiatives in the construction industry enhance the interoperability and efficiency of data exchange experienced in multiple kinds of research (Sharafat *et al.*, 2021; Shi, Pan, Jiang and Zhai, 2023; Shi, Pan, Jiang, Chen, *et al.*, 2023). ISO 15926 focuses on enabling data exchange related to various aspects of industrial plants, allowing different systems to share and integrate data (Kim *et al.*, 2020). However, no unique interoperability ecosystem is prone to integrating multiple stakeholders and systems (Doe *et al.*, 2022). Based on that, this work explores and tests an approach of Databus, where the users can request information over various types of systems or databases from an API that can search and integrate this data to return valuable information to the users.

BACKGROUND

The use of Linked Data (LD), or Web of Data, has been increasing in the last two decades cooperatively with the concept of the "Semantic Web" and aims to enable humans and computer systems to publish, share and connect the data on the Web (Poveda-Villalón, 2012; Vandenbussche *et al.*, 2016). This expansion was possible because the previous works on the Linked Data principles and Semantic Web (Berners-Lee, 2006) provided conditions to expand these technologies. To support the Linked Data and Semantic Web, a set of integrated technologies like Resource Description Framework (RDF) graphs, Uniform Resource Identifiers (URI), RDF Query Language (SPARQL), and the Web Ontology Language are standardised by the World Wide Web Consortium (W3C) (Pauwels *et al.*, 2018). Semantic web technologies are considered one of the most promising methods for achieving efficient information exchange because they can share data and improve interoperability among highly heterogeneous systems (Tchouanguem Djuedja *et al.*, 2021).

An RDF graph is a data model that uses semantic triples in the schema of subject–predicate–object expressions linked to each other via URIs (Pauwels and Terkaj, 2016; Pauwels *et al.*, 2018). It can be written in RDF/XML or different formats, such as Turtle or JSON-LD. Larger RDF datasets are stored in specialised databases called triple or quad stores for easy access, linking, and querying over networks (Pauwels and Terkaj, 2016; Pauwels *et al.*, 2018). For example, the construction industry uses the term Linked Building Data (LBD) to describe the implementation of semantic web technologies (SWT) for organising information over a collection of RDFs for multiple stakeholders over distinguished software and the web (Tchouanguem Djuedja *et al.*, 2021). While the Linked Building Data field is still in its infancy, it has already produced promising results that have garnered attention from industry professionals.

The IFC is an open standard data model of the buildingSMART standardisation organisation used for exchanging information in the Architecture, Engineering, Construction, Owner Operator (AECOO) industry and achieving interoperability as one of the BIM premises. Adding to this, some research on the industrial sector has started to apply the IFC for interoperability (D. L. de M. Nascimento, Roeder, Calvetti, Lopez, *et al.*, 2022; D. L. de M. Nascimento, Roeder, Calvetti, Mustelier, *et al.*, 2022; D. L. M. Nascimento *et al.*, 2022; Calvetti *et al.*, 2023). Also, Linked Open Data (LOD) and SWT are required and efficient tools to manage information in the construction business and support the data interoperability (Pauwels and Terkaj, 2016). This scenario led efforts to the development of the ifcOWL, the Ontology Web Language (OWL), to make the IFC available in RDF format (Beetz, van Leeuwen and de Vries, 2009; Pauwels *et al.*, 2015; Pauwels and Terkaj, 2016).

Semantic web technologies have been used to monitor industrial plants and to manage and exchange an enormous amount of data for supporting the design and operation phases (Terkaj, Tolio and Urgo, 2015). Also, promising results from analysing the use of Linked Data for building performance were obtained by combining the Internet of Things (IoT) technologies capable of reading Time-series data from various sensors and data storing in InfluxDB (Donkers *et al.*, 2021). The IoT concept allows physical objects to be interconnected through the integration of embedded electronics, software, and sensors (Lekic and Gardasevic, 2018). This connectivity enables these objects to exchange data with stakeholders and other connected devices (Lekic and Gardasevic, 2018).

The IoT enhances the ability to remotely monitor and manipulate physical entities via existing network infrastructure, thereby increasing efficiency, precision, and economic benefits while reducing human intervention. The incorporation of sensors into the IoT paradigm has given rise to intelligent cities, homes, power grids, virtual power plants, and transportation systems. This work presents technologies commonly used in IoT, such as MQTT, WebSockets, server-sent events for sending messages, and InfluxDB for storing Time-series data. MQTT (Message Queuing Telemetry Transport) was released in 1999 by IBM and is a lightweight messaging protocol that enables efficient communication between devices or applications with limited processing power and bandwidth (Soni and Makwana, 2017). It follows a publish-subscribe model, where publishers send messages on specific topics, and subscribers receive notifications on the issues of their interest. MQTT's small packet size and low overhead make it an ideal choice for IoT devices with limited memory, processing power, and network resources. Server-Sent Events and WebSocket tools are utilised for server-to-client interaction. Both facilitate the delivery of messages from the server to the client (browser), but WebSocket also allows for messages to be transmitted in the reverse direction (Słodziak and Nowak, 2016). InfluxDB is an open-source time-series database deployed by InfluxData that stores data as individual points consisting of a fieldset and timestamp (Kang and Hong, 2018). These points can be indexed or unindexed, providing advantages to CPU performance, disk space, and query response time. Its suitability for sensor monitoring cases with big data has made it a popular choice for time-series data storage.

Node-RED is a flexible tool for simulating the integration of these technologies. Node-RED is a freely available development tool that utilises a flow-based design to create prototypes and expedites the development of applications by integrating IoT hardware devices, Application Programming Interfaces (APIs), and online services (Lekic and Gardasevic, 2018). In order to combine and integrate the different types of information received from the sensors, an API is usually combined with a data treatment that filters and directs every kind of data to specific processing according to its typology. The monolithic Databus concept can be expanded to efficient microservices, thus increasing the system's efficiency. service is an architectural approach employed in enterprise applications that involves decomposing a large, monolithic application into numerous autonomous, smaller standalone components (Mütsch, 2016). This strategy enhances adaptability and scalability by developing fine-grained services that interact through lightweight mechanisms, often via an HTTP resource API (Mütsch, 2016).

Streaming data sources approach

The approach proposed in this paper is based on the premise that sensor time-series data can be searchable the same way any Knowledge Base information can, by publishing traditional time-series databases and stream endpoints in Linked Data format. This approach can offer several benefits. These include efficient data management, improved data accessibility, enhanced data integration, scalability, near real-time analysis, and data provenance and trust support. The Industrial Internet of Things (IIoT) enhances efficiency through industry by creating a secure data-sharing mechanism for sensors and actuators (Meng and Li, 2021). In machine manufacturing, digital services in the IIoT sector bring solutions like effective condition monitoring and predictive maintenance (Kammerer *et al.*, 2020). Data Interlinking, the process of connecting the published dataset with related data sources in the cloud, is a significant challenge in publishing the Linked Data (Martin, Kühl and Satzger, 2021). MQTT and Kafka can collaborate for near real-time bidirectional data processing (Choii *et al.*, 2010). These applications benefit from sharing sensor data via URIs and interoperability between MQTT data, RDF datasets, and IFC schema.

METHOD

The research demonstrates that the prototype fulfils its intended functions. The Proof of Concept (PoC) conducted comprises three main elements: (1) the research prototypes, (2) a proof of concept demonstration, which confirms that the prototypes deliver the anticipated outcomes, and (3) a post facto arguments, which advocate for the continued deployment of the prototypes targeting further developments (Elliott, 2021). The methodology suggested in this study advocates that data from sensor time series can be integrated into triple stores and accessed through an API. This prototype API can be set up to link the URIs to the appropriate data points. After, for demonstration, a system was developed using Node-Red with multiple data connection types and InfluxDB to implement and test a streaming sensor data point over URIs. The Node-Red is not designed to build highly scalable and collaborative backends - your best bet is to use it to create a rapid prototype first and then consider a language or framework, like Node.js, Nest or Golang, to plan tasks for the development team.

Finally, the findings of the search were conducted for comparable properties across several established ontologies, including Sensor Data simulator and storage, Time-series storage, Linked data endpoints, RDF store, and OWL ontologies design. Results and discussions highlight possible practical industrial uses. Further developments are envisaged by testing a new approach to query the SPARQL endpoint using an API. It suggests exploring data streaming and developing an IFC file generator for enhancing supply-chain data use. The importance of user-friendly approaches and system-to-system interaction for data interoperability is also highlighted.

Search for equivalent properties in existing ontologies

Reusing existing and consolidated properties is good practice instead of creating new ones whenever possible. So, before creating our properties, we conducted a search for existing ones with the aid of the Linked Open Vocabulary (https://lov.linkeddata.es/dataset/lov/) (*Linked Open Vocabularies (LOV)*, no date). Table 1 summarises the main ontologies found.

Table 1 - Linked Open Vocabulary searched

Property	Ontology	Applicable?	Reason
IfcTimeSeries (https://w3id.org/ifc/IFC4_ADD1#IfcTimeSeries)	https://pi.pauwel.be/evoc/ifc_W3ID, no date	No	A property in JSON format and not IFC will be required for this work.
TimeSeriesObservation (http://def.seegrid.csiro.au/isotc211/iso19156/2011/observation#TimeSeriesObservation)	http://def.seegrid.csiro.au, no date	No	This ontology is not accessible on the server.
hasTimeSeriesMesurement (https://saref.etsi.org/saref4ehaw/hasTimeSeriesMesurement)	https://saref.etsi.org/saref4ehaw/v1.1.1/, no date	No	It is unclear what is the data format for this property.
TimeSeriesMeasurement (https://saref.etsi.org/saref4ehaw/TimeSeriesMeasurement)	https://saref.etsi.org/saref4ehaw/v1.1.1/, no date	No	The purpose of this property is storing time series data in RDF, which is not the case of this work.
StreamEndpoint (https://riccardotommasini.github.io/vocals/docs/core/index-en.html#StreamEndpoint)	https://riccardotommasini.github.io, no date	No	Because this work proposes a property for each specific type of data stream technology.

Streaming data sources deployment

MQTT is a vital protocol for IIoT (Industrial Internet of Things) due to its lightweight nature, scalability, reliability, Quality of Service (QoS), and security features (Quincozes, Emilio and Kazienko, 2019). It is designed for low bandwidth and unstable network environments, making it ideal for real-time messaging services in network-connected devices (Salagean and Zinca, 2020; Roldán-Gómez *et al.*, 2022). MQTT can handle large-scale projects, ensuring message delivery even in unreliable networks (Salagean and Zinca, 2020; Roldán-Gómez *et al.*, 2022). It also supports various authentication and data security mechanisms (Salagean and Zinca, 2020). In the context of IIoT and enterprise layers, MQTT ensures efficient communication between IoT devices and the cloud (Quincozes, Emilio and Kazienko, 2019; Salagean and Zinca, 2020; Roldán-Gómez *et al.*, 2022).

Based on that, a system was created with a sensor simulator using the MQTT protocol and a time-series storage to deploy the sensor streaming data approach. The sensor simulator was built with a random integer numbers generator that, each second, publishes the values to an MQTT topic on an MQTT broker (MQTT server), see Figures 1 and 2. That will allow us to test the approach envisaged as a proof of concept (PoC). Over that, most industrial projects may scale up that PoC.

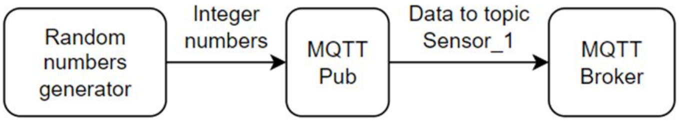


Figure 1 - Sensor simulator schema

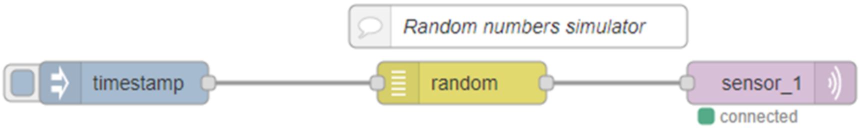


Figure 2 - Sensor simulator Node-red snippet schema

Database developed for the time-series

The storage receives data from MQTT and stores a measurement at an InfluxDB time-series database (DB) as presented in Figures 3 and 4.

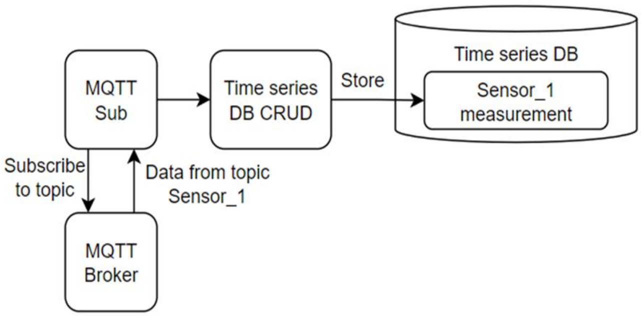


Figure 3 - Time-series storage schema

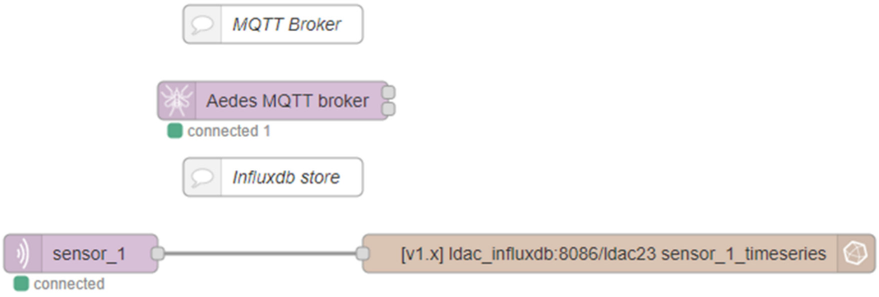


Figure 4 - Node-red snippet schema to time-series storage

This work proposes some linked data endpoints in the form of URIs as literals of triples of a sensor. The endpoints were developed to return data without knowledge of the internal storage architecture.

Implementation and test of the time-series data (PoC)

This GET REST API Endpoint queries the time-series database to get the sensor's current value. When the endpoint is reached, it gets the current timestamp and queries InfluxDB's last written value. It returns the sensor value if its timestamp is greater or equal to the timestamp when the endpoint is reached. An empty value is returned if it is lesser, meaning the last sensor value written to the database is outdated. This endpoint is accessed by accessing the URL <http://my_server/timeseries/Sensor_1?currentvalue>. Figure 5 presents the API schema, and Figure 6 the respective PoC.

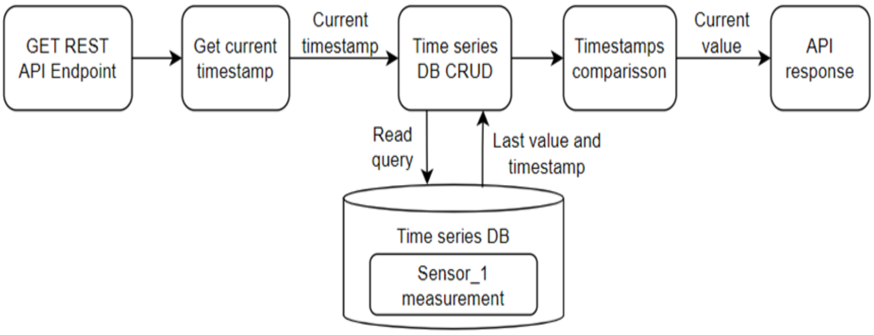


Figure 5 - API schema to get the current sensor value Endpoint

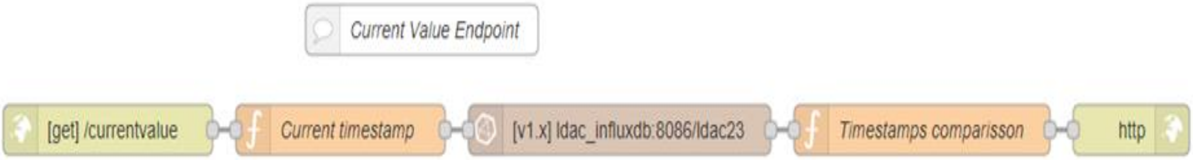


Figure 6 - Node-red snippet schema to get the current sensor value Endpoint

To get the last written value, another endpoint was developed (Figures 7 and 8). This GET REST API Endpoint queries the time-series database to get the sensor's latest value and timestamp. This endpoint is accessed by accessing the URL <http://my_server/timeseries/Sensor_1?lastvalue>; <http://my_server/timeseries/Sensor_1?lastvaluetimestamp>.

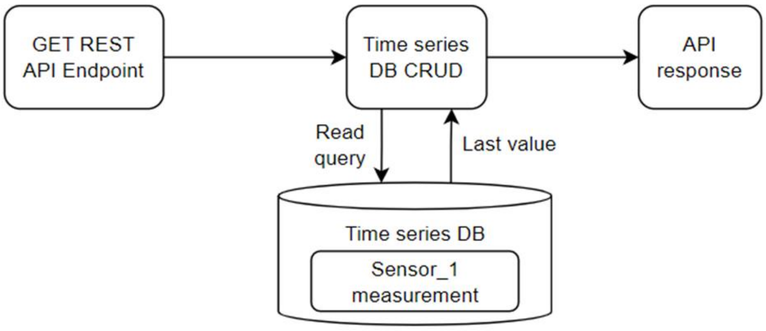


Figure 7 - API schema to get the sensor Last Value Endpoint

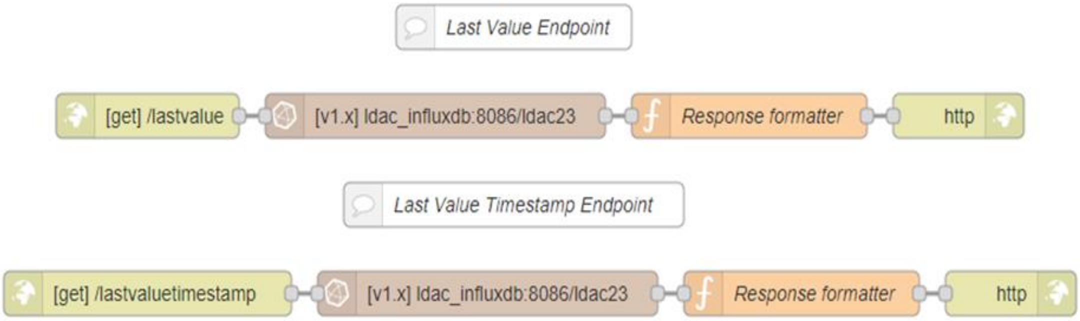

Last Value Endpoint
[get] /lastvalue
[v1.x] ldac_influxdb:8086/ldac23
Response formatter
http
Last Value Timestamp Endpoint
[get] /lastvaluetimestamp
[v1.x] ldac_influxdb:8086/ldac23
Response formatter
http

Figure 8 - Node-red snippet schema to get the sensor Last Value Endpoint

This GET REST API Endpoint queries the time-series database. The requester sends the sensor id and initial and end times in timestamp format, and the time-series DB CRUD queries the DB and creates a JSON payload to the API response. This endpoint is accessed by accessing the URL <http://my_server/timeseries/Sensor_1?initial_timestamp=0&final_timestamp=0>. Figure 9 presents the API schema, and Figure 10 the respective PoC.

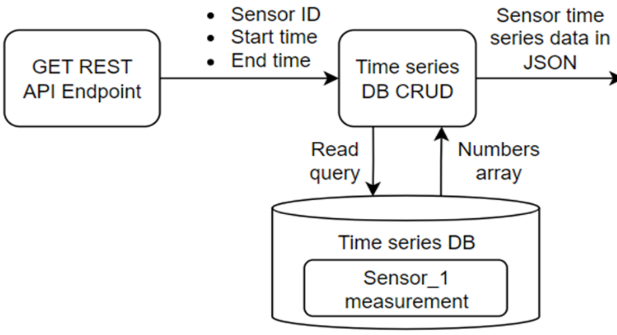


Figure 9 - API schema to get the sensor time-series Endpoint

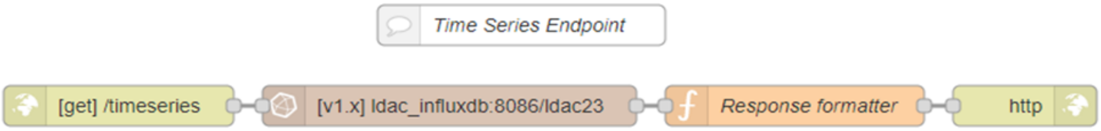


Figure 10 - Node-red snippet schema to get the sensor Time-series Endpoint

The Time-series URI thought so users can intuitively discover that they must fill the initial and final times in timestamp format. However, it cannot be accessed directly without user intervention, making it not user-friendly because the user should choose the initial and final timestamps and recreate the URI. A more user-friendly feature will be explored in future works. The URIs example is:
http://my_server/timeseries/Sensor_1?initial_timestamp=1644699969&final_timestamp=1647032769

This example's initial timestamp stands for Sat Feb 12 2022 21:06:09 GMT+0000, and the final timestamp stands for Fri Mar 11 2022 21:06:09 GMT+0000. A period of one month approximately one year ago.

In addition to the linked data endpoints, a static file endpoint was created for the IFC users. This GET REST API Endpoint queries the time-series and sensor IFC databases. The requester sends the sensor id and initial and end times in timestamp format, and the time-series DB CRUD queries the DB and creates a payload to the API response. Then, the IFC DB CRUD gets the sensor IFC project, and finally, an integrator makes the final file in IFC format.

For the proof of concept, when the endpoint is reached, we fetch a static IFC file and return it as the payload (Figures 11 and 12). Still, as an improvement, we plan to develop an automatic IFC file integrator to build this payload programmatically. This endpoint is accessed by accessing the URL <http://my_server/ifctimeseries/Sensor_1?initial_timestamp=0&final_timestamp=0>.

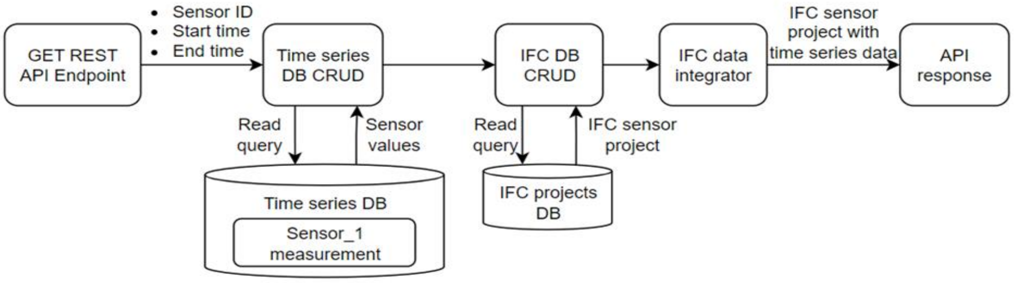


Figure 11 - API schema to get the sensor IFC time-series Endpoint

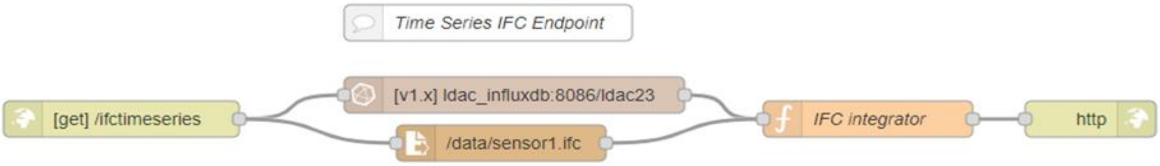


Figure 12 - Node-red snippet schema to get the sensor IFC Time-series Endpoint

Additionally, to the direct data access endpoints, two data stream endpoints were created: One using WebSocket technology and another using SSE (Server-Sent Events). Both automatically stream data with a single request connection. The WebSocket API Endpoint exposes the sensor data stream directly from the MQTT receiver, which is subscribed to the Sensor_1 topic. A WebSocket client should connect to this endpoint to get the stream of values. This endpoint is accessed by accessing the URL <ws://my_server/ws_datastream/Sensor_1>. Figure 9 presents the WebSocket API Endpoint schema, and Figure 10 the respective PoC.

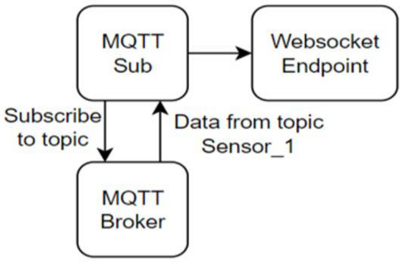


Figure 13 - WebSocket API Endpoint schema

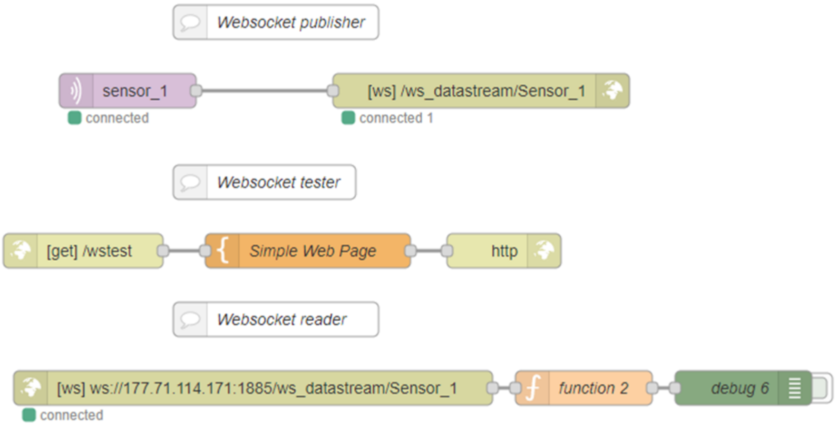


Figure 14 - Node-red snippet WebSocket API Endpoint schema

SSE stands for Server-Sent Events. The SSE API Endpoint exposes the sensor data stream directly from the MQTT receiver, which is subscribed to the Sensor_1 topic. An SSE client should connect to this endpoint to get the stream of values (Figures 15 and 16). This endpoint is accessed by accessing the URL <http://my_server/sse_datastream/Sensor_1>.

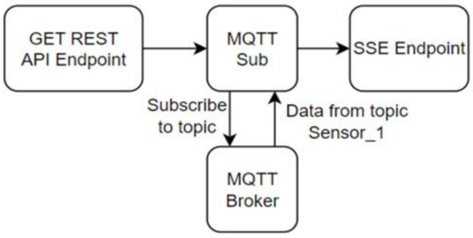


Figure 15 - SSE API Endpoint schema

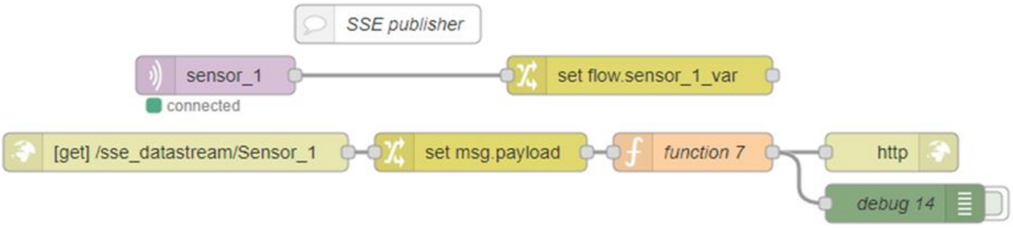
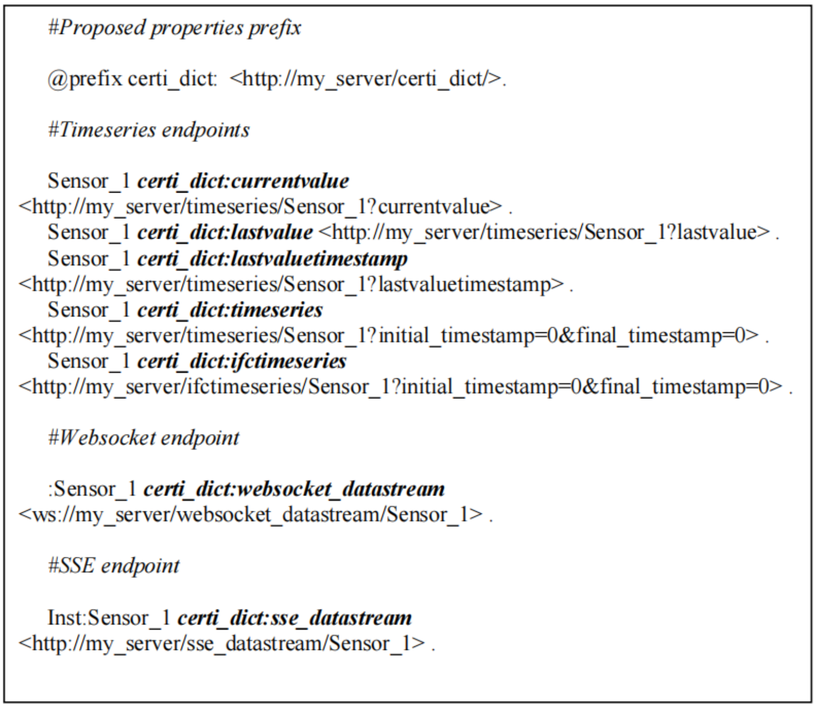


Figure 16 - Node-red snippet SSE API Endpoint schema

Implementation of the RDF store

The RDF store was implemented with Ontograph GraphDB, with the sensor and all the above endpoints in URI format as properties of the sensor. The properties were stored in a triple store according to this turtle RDF presented in Figure 17.



```
    #Proposed properties prefix

    @prefix certi_dict:  <http://my_server/certi_dict/>.

    #Timeseries endpoints

    Sensor_1 certi_dict:currentvalue
<http://my_server/timeseries/Sensor_1?currentvalue> .
    Sensor_1 certi_dict:lastvalue <http://my_server/timeseries/Sensor_1?lastvalue> .
    Sensor_1 certi_dict:lastvaluetimestamp
<http://my_server/timeseries/Sensor_1?lastvaluetimestamp> .
    Sensor_1 certi_dict:timeseries
<http://my_server/timeseries/Sensor_1?initial_timestamp=0&final_timestamp=0> .
    Sensor_1 certi_dict:ifctimeseries
<http://my_server/ifctimeseries/Sensor_1?initial_timestamp=0&final_timestamp=0> .

    #Websocket endpoint

    :Sensor_1 certi_dict:websocket_datastream
<ws://my_server/websocket_datastream/Sensor_1> .

    #SSE endpoint

    Inst:Sensor_1 certi_dict:sse_datastream
<http://my_server/sse_datastream/Sensor_1> .
```

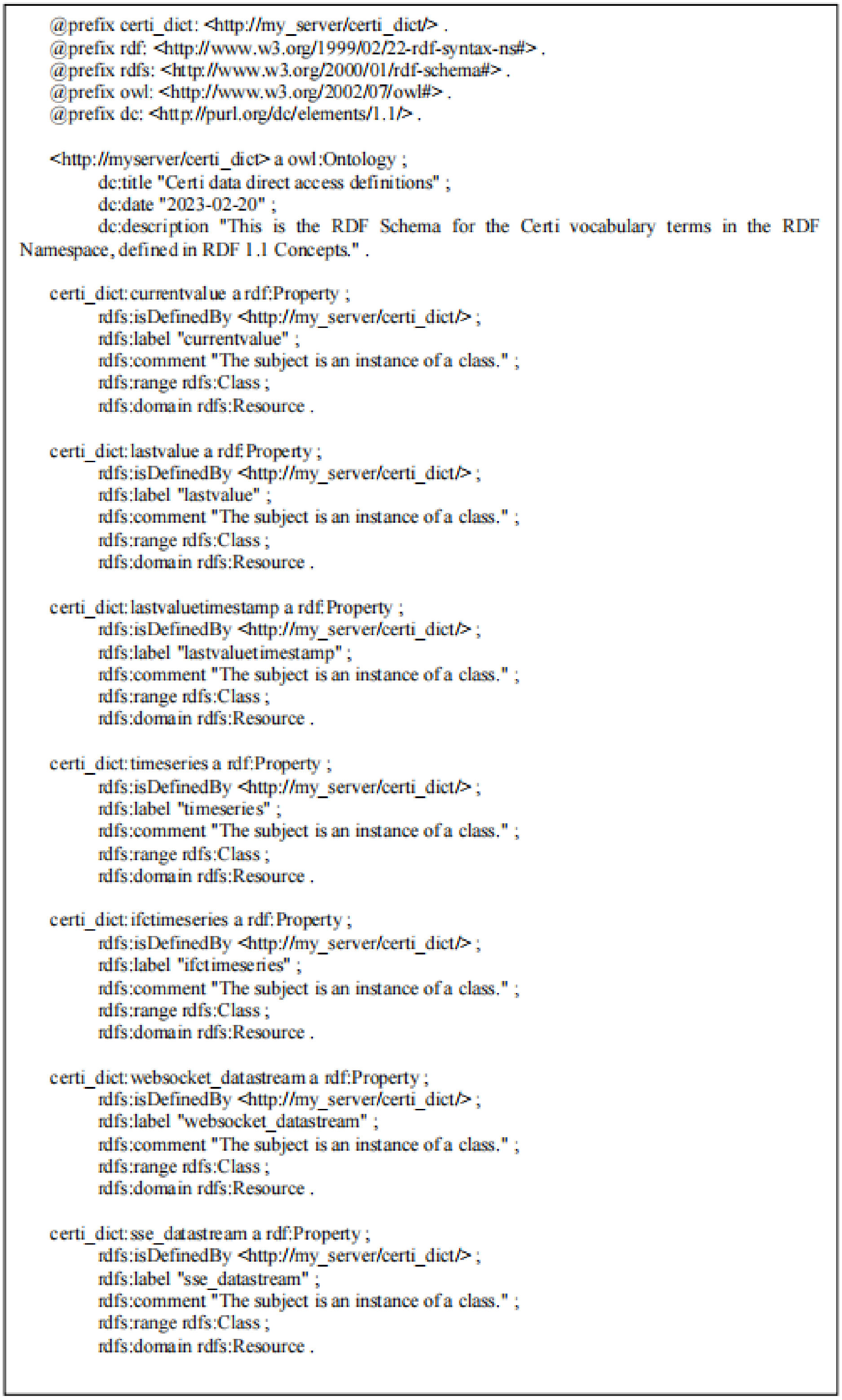
Figure 17 - Properties of the sensor according to turtle RDF

Determination of the ontology description (Certi_dict owl file)

This ontology provides a formal description of the properties used in the system to represent data and can be used to ensure consistent use of vocabulary terms in the application. Defines several properties that can be used to describe resources in the system. The properties defined in this ontology are:

- certi_dict:currentvalue
- certi_dict:lastvalue
- certi_dict:lastvaluetimestamp
- certi_dict:timeseries
- certi_dict:ifctimeseries
- certi_dict:websocket_datastream
- certi_dict:sse_datastream

Each property is defined to have a range of rdfs:Class, which means that it can be applied to any resource that is an instance of an RDF class. The properties are also defined to have a domain of rdfs:Resource, meaning they can be applied to any resource in the ontology, as shown in Figure 18.



```
    @prefix certi_dict: <http://my_server/certi_dict/> .
    @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
    @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
    @prefix owl: <http://www.w3.org/2002/07/owl#> .
    @prefix dc: <http://purl.org/dc/elements/1.1/> .

    <http://myserver/certi_dict> a owl:Ontology ;
            dc:title "Certi data direct access definitions" ;
            dc:date "2023-02-20" ;
            dc:description "This is the RDF Schema for the Certi vocabulary terms in the RDF
Namespace, defined in RDF 1.1 Concepts." .

    certi_dict:currentvalue a rdf:Property ;
            rdfs:isDefinedBy <http://my_server/certi_dict/> ;
            rdfs:label "currentvalue" ;
            rdfs:comment "The subject is an instance of a class." ;
            rdfs:range rdfs:Class ;
            rdfs:domain rdfs:Resource .

    certi_dict:lastvalue a rdf:Property ;
            rdfs:isDefinedBy <http://my_server/certi_dict/> ;
            rdfs:label "lastvalue" ;
            rdfs:comment "The subject is an instance of a class." ;
            rdfs:range rdfs:Class ;
            rdfs:domain rdfs:Resource .

    certi_dict:lastvaluetimestamp a rdf:Property ;
            rdfs:isDefinedBy <http://my_server/certi_dict/> ;
            rdfs:label "lastvaluetimestamp" ;
            rdfs:comment "The subject is an instance of a class." ;
            rdfs:range rdfs:Class ;
            rdfs:domain rdfs:Resource .

    certi_dict:timeseries a rdf:Property ;
            rdfs:isDefinedBy <http://my_server/certi_dict/> ;
            rdfs:label "timeseries" ;
            rdfs:comment "The subject is an instance of a class." ;
            rdfs:range rdfs:Class ;
            rdfs:domain rdfs:Resource .

    certi_dict:ifctimeseries a rdf:Property ;
            rdfs:isDefinedBy <http://my_server/certi_dict/> ;
            rdfs:label "ifctimeseries" ;
            rdfs:comment "The subject is an instance of a class." ;
            rdfs:range rdfs:Class ;
            rdfs:domain rdfs:Resource .

    certi_dict:websocket_datastream a rdf:Property ;
            rdfs:isDefinedBy <http://my_server/certi_dict/> ;
            rdfs:label "websocket_datastream" ;
            rdfs:comment "The subject is an instance of a class." ;
            rdfs:range rdfs:Class ;
            rdfs:domain rdfs:Resource .

    certi_dict:sse_datastream a rdf:Property ;
            rdfs:isDefinedBy <http://my_server/certi_dict/> ;
            rdfs:label "sse_datastream" ;
            rdfs:comment "The subject is an instance of a class." ;
            rdfs:range rdfs:Class ;
            rdfs:domain rdfs:Resource .
```

Figure 18 - RDF Schema for the Certi Vocabulary Terms

RESULTS AND DISCUSSION

The Linked Data spectrum part of this proof of concept (PoC) was designed to access endpoints without user authentication, i.e., it works only on a local network. Typically, these endpoints cannot be opened to the Internet on systems with private business data. In future work, this system will be improved with user authentication to access endpoints over the Internet. This work created a user-intuitive URI with the suffix ?initial_timestamp=0&final_timestamp=0; however, it is not machine-readable yet. In this PoC, the machine would have to mount the initial and end times in the URI automatically. In future works, this feature will be explored and enhanced.

A 3D representation is relevant for users' visualisation and engagement. When a user or a machine accesses the triple store, it will "see" the proposed endpoints as properties of the sensor. The primary possibility of use would be visual analyses and data interaction. A 3D viewer could infer that the "has data stream" property is associated with the equipment sensor and then use rules to show equipment statistics in different colours depending on the measured sensor value, for example.

Also, to support users' data use, dashboard software like Grafana could read the triple store and show real-time sensor data on its dashboards. It is also possible to build a Grafana plugin to read triple stores. However, Grafana could access the WebSocket endpoint directly and get the sensor values stream. Program alarms, get metrics, and indicators are information users' demand. The RDF could store metrics properties to the sensor value to create alarms based on the sensor's real-time values. For example, a piece of equipment has a chance to explode if its temperature is up to 500°C. Suppose the equipment related to the sensor has a property like "maximum temperature" filled with 500°C when the live sensor data gets above 500°C. In that case, a reasoner could automatically infer that the equipment is at risk. Complex metrics and indicators could be constructed with the same idea. Over users' interaction, it will be explored in future works.

The APIs developed allow systems-to-systems interaction, proving the approach valuable. In the future, an API can also fetch the data in different graph databases, such as GraphDB (RDF graph) and Neo4J (property graph), extending the same API concept that implements a data bus with multiple data sources used in this work. Users' post-analyses will require time-series collection. Based on that, another intention is to build an automatic IfcTimeSeries. The requester sends the sensor "id" and initial and end times in timestamp format, and the time-series DB CRUD queries the time-series DB, creates the IfcTimeSeries variable, queries the IFC DB to get the sensor IFC file, and incorporates the time-series values to the IFC file.

Its research is scalable mainly because it is based on Linked Data (LD). Over that, it increases semantic queries. Linked Data allows data interconnection across different sources, providing a more comprehensive and interconnected view of information. This can significantly improve the accuracy and relevance of semantic queries. Also, data availability may enhance the company's Learning Organisation (LO) (Tortorella *et al.*, 2019). However, many current Internet of Things (IoT) systems lack integration with Resource Description Framework (RDF) and LD. This can limit the ability of these systems to leverage the benefits of semantic queries and LD entirely. Integrating LD over RDF with IIoT may open new paths for data interaction and semantic queries. It could empower systems to respond to complex queries, enhancing their functionality and usability.

Using LD over semantic queries may provide significant benefits in terms of cost and time. Connected to increase efficiency, LD allows data interconnection across different sources, improving the accuracy and relevance of semantic data and reducing the time and effort required to gather and analyse data (Konstantinou and Spanos, 2015). For example, accurate data for facility management connected with BIM and Lean may enhance collaboration across the management process (Nascimento *et al.*, 2018). Also, LD is highly scalable and very suitable for dealing with big data, leading to cost savings in data storage and processing (Konstantinou and Spanos, 2015). Adding to this, using Web technologies such as HTTP and URIs makes integrating with existing systems easier and reduces the costs associated with data integration (Konstantinou and Spanos, 2015). Finally, LD may support more informed decision-making by providing a more comprehensive view of data, leading to saving costs. However, it is crucial to notice that the actual price and time savings will depend on different factors, such as the data quality, the technology infrastructure and the use case scenarios.

CONCLUSION

This paper discusses the importance of data sensors in industrial operations for ensuring safety and improving production. It emphasises the need for friendly access to operating data in the context of the industry 4.0 scenario. The work presented introduces the concept of using valid data points over URIs for sensor time-series and storing them in triple stores via APIs. The approach involves integrating industrial sensor data and sharing it via URIs. The proof of concept presented demonstrates the feasibility of sharing sensor data via URIs and lays the foundation for interoperability between MQTT data, RDF datasets, and IFC schema. The investigation's limitations include using simulated data, but it is expected to yield similar results with real data. The proposed solution has practical implications for companies looking to enhance data shareability and interoperability and researchers interested in advancing the solution for specific features. The paper offers insights into future deployments of systems-to-systems interoperability, focusing on user-friendly data shareability, and contributes to both industrial and academic developments.

Users might be limited to standard access and topic-based inquiries without the ease of URLs and semantic searches. This could potentially limit the scope and efficiency of conducting searches. This research was intended as a structuring basis for future uses. The PoC is a starting point for developers to structure semantic systems and develop more automated ones. By structuring semantic systems based on these models, developers may create computerised systems that understand and respond to complex queries. This can be particularly useful for advanced users who must perform semantic searches for everyday uses. Large language models (LLMs) can help overcome these limitations by providing a more nuanced understanding of language, enabling more precise and relevant search results. This could revolutionise the way users interact with data in their daily routines.

Further, a different approach will be tested to query the SPARQL endpoint (or create a GraphQL like it was proposed in this reference). The URI reaches an API that retrieves the time series values for the period and programmatically makes the time-series values as RDF triples. Finally, it will be beneficial to search for ways to stream data through the WebSocket endpoint and time-series payload from the time-series endpoint to be accessed directly by SPARQL queries for automatic data analysis. The IFC interoperability is mandatory for enhancing supply-chain data use and engagement. An IFC file generator can be developed, taking the result of a SPARQL query with one or more types of equipment and its sensors, with their properties, including time-series data, building the IFC file and aggregating to an existing 3D IFC file. It would be advantageous to users who want an IFC file to load in a 3D viewer. Concluding, user-friendly approaches and systems-to-systems interaction are critical elements of data interoperability.

ACKNOWLEDGMENTS

The part of this work conducted by Calvetti was done within the auspices of Base Funding UIDB/04708/2020 of the CONSTRUCT - Instituto de I&D em Estruturas e Construções - funded by national funds through the FCT/MCTES (PIDDAC).

REFERENCES

Beetz, J., van Leeuwen, J. and de Vries, B. (2009), "IfcOWL: A case of transforming EXPRESS schemas into ontologies", *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Vol. 23, No. 1, pp. 89–101. Available at: https://doi.org/10.1017/S0890060409000122.

Berners-Lee, T. (2006), "Linked Data - Design Issues", Available at: https://www.w3.org/DesignIssues/LinkedData.html (Accessed: 27 February 2023).

Calvetti, D. et al. (2023), "Interoperability framework for subsea sensors data", in *2023 European Conference on Computing in Construction and the 40th International CIB W78 Conference*. Available at: https://doi.org/10.35490/EC3.2023.259.

Choii, M.-Y. et al. (2010), "Interoperability between a relational data model and an RDF data model", in *The 6th International Conference on Networked Computing and Advanced Information Management*. pp. 335–340.

Daquino, M. et al. (2022), "Creating RESTful APIs over SPARQL endpoints using RAMOSE", *Semantic Web*, Vol. 13, No. 2, pp. 195–213. Available at: https://doi.org/10.3233/SW-210439.

Doe, R. et al. (2022), "Interoperability in AECO and the oil & gas sectors: object-based standards and systems", *Journal of Information Technology in Construction*, Vol. 27, pp. 312–334. Available at: https://doi.org/10.36680/j.itcon.2022.016.

Dojchinovski, M. et al. (2021), "The DBpedia Technology Tutorial", in *2021 Workshops and Tutorials - Language Data and Knowledge, LDK 2021*.

Donkers, A. et al. (2021), "Real-Time Building Performance Monitoring using Semantic Digital Twins", in *CEUR Workshop Proceedings. 9th Linked Data in Architecture and Construction Workshop, LDAC 2021*, Luxembourg, pp. 55–66.

Elliott, S. (2021), "Proof of Concept Research", *Philosophy of Science*, Vol. 88, No. 2, pp. 258–280. Available at: https://doi.org/10.1086/711503.

Frey, J. and Hellmann, S. (2021), "FAIR Linked Data - Towards a Linked Data Backbone for Users and Machines", in *Companion Proceedings of the Web Conference 2021. WWW '21: The Web Conference 2021*, Ljubljana Slovenia: ACM, pp. 431–435. Available at: https://doi.org/10.1145/3442442.3451364.

Hernawan, A. et al. (2021), "A generalization SPARQL federated query: An initial step towards machine-readable web of data for halal food products", *IOP Conference Series: Earth and Environmental Science*, Vol. 913, No. 1, p. 012040. Available at: https://doi.org/10.1088/1755-1315/913/1/012040.

Javaid, M. et al. (2021), "Significance of sensors for industry 4.0: Roles, capabilities, and applications", *Sensors International*, Vol. 2, p. 100110. Available at:

https://doi.org/10.1016/j.sintl.2021.100110.

Kalsoom, T. et al. (2020), "Advances in Sensor Technologies in the Era of Smart Factory and Industry 4.0", *Sensors*, Vol. 20, No. 23, p. 6783. Available at: https://doi.org/10.3390/s20236783.

Kammerer, K. et al. (2020), "Process-Driven and Flow-Based Processing of Industrial Sensor Data", *Sensors*, Vol. 20, No. 18, p. 5245. Available at: https://doi.org/10.3390/s20185245.

Kang, T.W. and Hong, C.H. (2018), "IFC-CityGML LOD mapping automation using multiprocessing-based screen-buffer scanning including mapping rule", *KSCE Journal of Civil Engineering*, Vol. 22, No. 2, pp. 373–383. Available at: https://doi.org/10.1007/s12205-017-0595-9.

Kenda, K. et al. (2019), "Streaming Data Fusion for the Internet of Things", *Sensors*, Vol. 19, No. 8, p. 1955. Available at: https://doi.org/10.3390/s19081955.

Kim, B.C. et al. (2020), "ISO 15926–based integration of process plant life-cycle information including maintenance activity", *Concurrent Engineering*, Vol. 28, No. 1, pp. 58–71. Available at: https://doi.org/10.1177/1063293X19894041.

Konstantinou, N. and Spanos, D.-E. (2015), "Introduction: Linked Data and the Semantic Web", in Konstantinou, N. and Spanos, D.-E., *Materializing the Web of Linked Data*. Cham: Springer International Publishing, pp. 1–16. Available at: https://doi.org/10.1007/978-3-319-16074-0_1.

Lekic, M. and Gardasevic, G. (2018), "IoT sensor integration to Node-RED platform", in *2018 17th International Symposium INFOTEH-JAHORINA (INFOTEH)*. East Sarajevo: IEEE, pp. 1–5. Available at: https://doi.org/10.1109/INFOTEH.2018.8345544.

Linked Open Vocabularies (LOV) (no date), Available at: https://lov.linkeddata.es/dataset/lov/ (Accessed: 13 November 2023).

Martin, D., Kühl, N. and Satzger, G. (2021), "Virtual Sensors", *Business & Information Systems Engineering*, Vol. 63, No. 3, pp. 315–323. Available at: https://doi.org/10.1007/s12599-021-00689-w.

Meng, Y. and Li, J. (2021), "Data Sharing Mechanism of Sensors and Actuators of Industrial IoT Based on Blockchain-Assisted Identity-Based Cryptography", *Sensors*, Vol. 21, No. 18, p. 6084. Available at: https://doi.org/10.3390/s21186084.

Mütsch, F. (2016), *Design of a Linked Data-enabled Microservice Platform for the Industrial Internet of Things*. Karlsruher Institut für Technologie.

Nascimento, D.L. de M., Roeder, A.B., Calvetti, D., Lopez, A.C., et al. (2022), "Digital Commissioning Processes for the Oil and Gas Sector", in *39th International Symposium on Automation and Robotics in Construction*. Available at: https://doi.org/10.22260/ISARC2022/0073.

Nascimento, D.L. de M., Roeder, A.B., Calvetti, D., Mustelier, A.M., et al. (2022), "Industrial Plants Commissioning Procedures Using Digital Process-Oriented Architecture", *IFAC-PapersOnLine*, Vol. 55, No. 10, pp. 2079–2084. Available at: https://doi.org/10.1016/j.ifacol.2022.10.014.

Nascimento, D.L.D.M. et al. (2018), "Facility management using digital Obeya room by integrating bim-lean approaches – an empirical study", *Journal of Civil Engineering and Management*, Vol. 24, No. 8, pp. 581–591. Available at: https://doi.org/10.3846/jcem.2018.5609.

Nascimento, D.L.M. et al. (2022), "Digitalisation initiative of O&G offshore projects contractual procedures", in *14th European Conference on Product and Process Modelling. ECPPM 2022*, Trondheim, Norway, p. 7. Available at: https://doi.org/10.1201/9781003354222-77.

Pauwels, P. et al. (2015), "Linked Data in Architecture and Construction", *Automation in Construction*, Vol. 57, pp. 175–177. Available at: https://doi.org/10.1016/j.autcon.2015.06.007.

Pauwels, P. et al. (2018), "Linked Data", in A. Borrmann et al. (eds) *Building Information Modeling*. Cham: Springer International Publishing, pp. 181–197. Available at: https://doi.org/10.1007/978-3-319-92862-3_10.

Pauwels, P. and Terkaj, W. (2016), "EXPRESS to OWL for construction industry: Towards a recommendable and usable ifcOWL ontology", *Automation in Construction*, Vol. 63, pp. 100–133. Available at: https://doi.org/10.1016/j.autcon.2015.12.003.

Poveda-Villalón, M. (2012), "A Reuse-Based Lightweight Method for Developing Linked Data Ontologies and Vocabularies", in E. Simperl et al. (eds) *The Semantic Web: Research and Applications*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 833–837.

Quincozes, S., Emilio, T. and Kazienko, J. (2019), "MQTT Protocol: Fundamentals, Tools and Future Directions", *IEEE Latin America Transactions*, Vol. 17, No. 09, pp. 1439–1448. Available at: https://doi.org/10.1109/TLA.2019.8931137.

Rehman, S.U. (2023), "Industry 4.0 adoption and firm efficiency: evidence from emerging Giants in

Asia Pacific region", *Brazilian Journal of Operations & Production Management*, Vol. 20, No. 3, p. 1958. Available at: https://doi.org/10.14488/BJOPM.1958.2023.

Roldán-Gómez, J. et al. (2022), "Security Analysis of the MQTT-SN Protocol for the Internet of Things", *Applied Sciences*, Vol. 12, No. 21, p. 10991. Available at: https://doi.org/10.3390/app122110991.

Salagean, M. and Zinca, D. (2020), "IoT Applications based on MQTT Protocol", in *2020 International Symposium on Electronics and Telecommunications (ISETC)*. Timisoara, Romania: IEEE, pp. 1–4. Available at: https://doi.org/10.1109/ISETC50328.2020.9301055.

Schery, C.A.D. et al. (2023), "BIM critical factors and benefits for public sector: from a systematic review to an empirical fuzzy multicriteria approach", *Brazilian Journal of Operations & Production Management*, Vol. 20, No. 3, p. 1837. Available at: https://doi.org/10.14488/BJOPM.1837.2023.

Sharafat, A. et al. (2021), "BIM-Based Tunnel Information Modeling Framework for Visualization, Management, and Simulation of Drill-and-Blast Tunneling Projects", *Journal of Computing in Civil Engineering*, Vol. 35, No. 2, p. 04020068. Available at: https://doi.org/10.1061/(ASCE)CP.1943-5487.0000955.

Shi, J., Pan, Z., Jiang, L. and Zhai, X. (2023), "An ontology-based methodology to establish city information model of digital twin city by merging BIM, GIS and IoT", *Advanced Engineering Informatics*, Vol. 57, p. 102114. Available at: https://doi.org/10.1016/j.aei.2023.102114.

Shi, J., Pan, Z., Jiang, L., Chen, P., et al. (2023), "Research on a methodology for intelligent seismic performance evaluation and optimization design of buildings based on IFC and ontology", *Engineering Structures*, Vol. 288, p. 116213. Available at: https://doi.org/10.1016/j.engstruct.2023.116213.

Słodziak, W. and Nowak, Z. (2016), "Performance Analysis of Web Systems Based on XMLHttpRequest, Server-Sent Events and WebSocket", in A. Grzech et al. (eds) *Information Systems Architecture and Technology: Proceedings of 36th International Conference on Information Systems Architecture and Technology – ISAT 2015 – Part II*. Cham: Springer International Publishing (Advances in Intelligent Systems and Computing), pp. 71–83. Available at: https://doi.org/10.1007/978-3-319-28561-0_6.

Soni, D. and Makwana, A. (2017), "A SURVEY ON MQTT: A PROTOCOL OF INTERNET OF THINGS(IOT)", in *International conference on telecommunication, power analysis and computing techniques (ICTPACT-2017)*.

Tchouanguem Djuedja, J.F. et al. (2021), "An integrated Linked Building Data system: AEC industry case", *Advances in Engineering Software*, Vol. 152, p. 102930. Available at: https://doi.org/10.1016/j.advengsoft.2020.102930.

Terkaj, W., Tolio, T. and Urgo, M. (2015), "A virtual factory approach for in situ simulation to support production and maintenance planning", *CIRP Annals*, Vol. 64, No. 1, pp. 451–454. Available at: https://doi.org/10.1016/j.cirp.2015.04.121.

Tortorella, G. et al. (2019), "How do different generations contribute to the development of a learning organization in companies undergoing a lean production implementation?", *The Learning Organization*, Vol. 27, No. 2, pp. 101–115. Available at: https://doi.org/10.1108/TLO-02-2019-0035.

Vandenbussche, P.-Y. et al. (2016), "Linked Open Vocabularies (LOV): A gateway to reusable semantic vocabularies on the Web", *Semantic Web*. Edited by M. Dumontier, Vol. 8, No. 3, pp. 437–452. Available at: https://doi.org/10.3233/SW-160213.

Vega-Gorgojo, G. (2022), "CRAFTS: Configurable REST APIs for Triple Stores", *IEEE Access*, Vol. 10, pp. 32426–32441. Available at: https://doi.org/10.1109/ACCESS.2022.3160610.

Author contributions: All authors contributed equally to this paper.