frontiers in Computer Science

ORIGINAL RESEARCH
published: 30 October 2020
doi: 10.3389/fcomp.2020.578673

Ontology-Based Context Modeling in Physical Asset Integrity Management

Ali Al-Shdifat, Christos Emmanouilidis*†, Muhammad Khan and Andrew G. Starr*

School of Aerospace, Transport and Manufacturing, Cranfield University, Cranfield, United Kingdom

Asset management is concerned with the management practices, technologies and tools necessary to maximize the value delivered by physical engineering assets. IoT-generated data are increasingly considered as an asset and the data asset value needs to be maximized too. However, asset-generated data in practice are often collected in non-actionable form. Collected data may comprise a wide number of parameters, over long periods of time and be of significant scale. Yet they may fail to represent the range of possible scenarios of asset operation or the causal relationships between the monitored parameters, and so the size of the data collection, while adding to the complexity of the problem, does not necessarily allow direct data asset value exploitation. One way to handle data complexity is to introduce context information modeling and management, wherein data and service delivery are determined upon resolving the apparent context of a service or data request. The aim of the present paper is, therefore, 2-fold: to analyse current approaches to addressing IoT context information management, mapping how context-aware computing addresses key challenges and supports the delivery of monitoring solutions; and to develop a maintenance context ontology focused on failure analysis of mechanical components so as to drive monitoring services adaptation. The approach is demonstrated by applying the ontology on an industrially relevant physical gearbox test rig, designed to model complex misalignment cases met in manufacturing and aerospace applications.

Keywords: internet of things, context information management, maintenance ontology, context sharing, physical asset management

OPEN ACCESS

Edited by:
Oscar Lazaro,
Innovalia Association, Spain

Reviewed by:
Laura Belli,
University of Parma, Italy
Luca Davoli,
University of Parma, Italy

****Correspondence:***
Ali Al-Shdifat
ali.alshdifat@cranfield.ac.uk
Christos Emmanouilidis
christosem@ieee.org

†Present address:
Christos Emmanouilidis,
Faculty of Economics and Business Operations — Management, Groningen, Netherlands

Specialty section:
This article was submitted to Mobile and Ubiquitous Computing, a section of the journal Frontiers in Computer Science

Received: *30 June 2020*
Accepted: *28 September 2020*
Published: *30 October 2020*

Citation:
Al-Shdifat A, Emmanouilidis C, Khan M and Starr AG (2020) Ontology-Based Context Modeling in Physical Asset Integrity Management. Front. Comput. Sci. 2:578673. doi: 10.3389/fcomp.2020.578673

INTRODUCTION

Typical applications of internet of things (IoT) technologies amalgamate the ability to identify, sense, compute, communicate and sometimes actuate, for the purpose of monitoring and remotely controlling the environment (de Matos et al., 2020). According to a Statistica report. (2020), it is predicted that the amount of devices with Internet connectivity will exceed 50 billion by 2030. Such devices produce significant volumes of data which are communicated through networks, and upon processing enable better informed decision making and actions. One method of handling the high complexity of such volumes of data is by introducing context information management. Context is a key aspect in the process of leveraging information concerning situations and enabling applications to be adapted according to the perceived context (Pradeep and Krishnamoorthy, 2019).

Systems with context awareness are employed within IoT environments for the purpose of sensing the operational environment and for delivering an appropriate response to both the user and application (Perera et al., 2014). Such systems are capable of analyzing the data generated by IoT devices, generating a high-level of semantic organization of data and then converting it into

context information. This information is subsequently utilized in determining an environment's status so as to drive appropriate responses. In general, the status of the environment is determined by a combination of circumstances, including users, applications, location, or devices (Abowd et al., 1999), which constitute the context information. As IoT technologies become more embedded in monitoring activities, there is a growing necessity to manage their context information in industrial environments. This entails gathering, modeling, reasoning, and disseminating context in order to efficiently manage the data generated by multiple devices and to ensure that they can be effectively integrated into enterprise systems. Nonetheless, data contributing to context information are often modeled or processed within the narrow scope of isolated subsystems, restricting interoperability. Moreover, even when similar systems for collecting context are applied in distinct settings, information is infrequently shared between them (Perera et al., 2014).

The ability to share context among different applications is a critical necessity for the IoT, making data shared between heterogeneous systems reusable in multiple applications (Ramachandran and Krishnamachari, 2019). Context information management has been recognized as a challenge for relevant research and early on Bernardos et al. (2008) developed a data fusion framework for context-awareness systems that included the following stages: (i) Obtaining context, (ii) processing context, (iii) reasoning and decision-making. Perttunen et al. (2009) have surveyed popular context reasoning and representation techniques and provided an overview of the requirements for context representation, arguing that such requirements were insufficiently covered in the literature regarding the interplay between efficiency, expressiveness, soundness, and completeness, with ontology-based approaches achieved improved scalability and reuse compared to other approaches. This finding is consistent with that of Bettini et al. (2010), although scalability of on-line reasoning with a large number of entities is raised as a significant challenge. This is the case when dealing with data of significant complexity and scale, as typically encountered in IoT applications (2020), making it important that the semantics of IoT data are captured by appropriate context modeling to gain valuable insights (Perera et al., 2014).

Context information management has largely dealt with the challenges of ubiquitous environments, as well as the data heterogeneity and services scalability. Nonetheless, while substantial research efforts have been devoted to context information management in web-based, mobile and ubiquitous computing, including IoT-enabled computing, little attention has been given to translate these advances to tangible progress in industrial monitoring services (Al-shdifat and Emmanouilidis, 2018). Context modeling in the literature is typically handled *via* ontologies. However, when dealing with monitoring services in manufacturing environments, developed approaches often lack expressiveness concerning the representation of the domain knowledge. To address such needs, this paper analyses requirements and produces a design for the components required to develop effective context-aware systems to enhance monitoring services in industrial environments. It then presents the development of a context resolution service focused on failure analysis of mechanical components so as to drive monitoring services adaptation. The paper is structured as follows. Section related work briefly discusses literature related to context information sharing and ontologies in maintenance and asset management. Section system framework and methodology presents the system framework and the ontology development process, based on established practice and maintenance vocabulary standards. An instantiation of the developed ontology is implemented for testing on an industrially relevant test rig and is presented in section implementation on a case study. Section results and discussion presents and discusses the ontology design and its implementation, including examples of context resolution results. The final section summarizes the key contributions of the paper and suggests potential further research pathways.

RELATED WORK

The following section presents a discussion on the basic concepts in the field of context-aware systems including context, context-awareness in IoT, the context information sharing, as well as the ontologies in maintenance and asset management.

Context Information Management

The concept "context-aware" system was originally proposed by Schilit and Theimer in 1994 stating that "A system is context-aware, if it uses context to provide relevant information and/or services to the user." Other early works have defined context as "any information that can be used to characterize the situation of an entity, an entity is a person, place, or object that is considered relevant to the interaction between a user and an application" (Abowd et al., 1999). Abowd and Mynatt (2000) specified the basic elements required for analyzing and understanding context, namely the five Ws (what, who, where, why, and when). According to Byun and Cheverst (2004), a system is defined as being context aware if it is capable of extracting, interpreting and using context information and its functionality can be adapted to the prevailing context of use. In the domain of asset and maintenance management, the early definition of context by Abowd et al. (1999) can be adopted and extended by specifying that context is relevant to the asset and its hierarchy, the user, the production or service business circumstances, as well as overall system and operating environment aspects (Emmanouilidis et al., 2019). Despite the generally acknowledged definitions of what is regarded as context awareness, a standard format and representation of the concept has not been established (Xu et al., 2014; Perera et al., 2015; de Matos et al., 2017). Various researchers have determined different typologies of context. Abowd et al. (1999) differentiated between primary and secondary context, in addition to conceptual and operational. Liu et al. (2011) stated that context can be classified as user, physical or networking. **Table 1** provides a summary of the different approaches adopted for categorizing context.

According to Perera et al. (2015), the steps required for a system to deliver context information are acquisition, modeling, reasoning, and distribution which combined form the context

TABLE 1 | Different context categorization schemes.

References	Context categorization
Abowd et al. (1999)	Who, Where, When What, and Why
Chen and Kotz (2000)	User, Computing, Physical, and When
Henricksen (2003)	Sensed, Static, Profiled, and Derived
van Bunningen et al. (2005)	Operational and Conceptual
Chong et al. (2007)	Computing, Physical, Historical, and Sensor
Rizou et al. (2010)	Observable and Non-Observable
Liu et al. (2011)	User, Physical, and Networking
Emmanouilidis et al. (2013)	User, Environment, System, Social, Service
Valverde-Rebaza et al. (2018)	Location and Social

lifecycle. In the acquisition step, the raw data are collected from sensors, databases or the surrounding environment. In the modeling stage, the data is brought into a particular representation so it can be converted into input for the reasoning stage. Various approaches have been described in the current literature for the modeling process, including key-value pairs, ontology, and markup schemes (Bettini et al., 2010; Snidaro et al., 2015). The semantic processing stage in the context lifecycle is the reasoning process, where different methods can be used for inferring context, such as such as supervised/unsupervised learning, rules, ontologies, probabilistic approaches, as well as data aggregation and fusion mechanisms (Perera et al., 2015). Hence, the context-awareness of a system is determined by its ability to utilize the context acquired via the context lifecycle to deliver beneficial information/services to users (Abowd et al., 1999). Several context-aware systems utilize context purely for decision support/making or direct distribution to the end user. Nevertheless, certain systems allow context information to be shared with other interested actors or subsystems. This is defined as context information sharing and represents one of key challenges in the field of context-awareness for IoT (Perera et al., 2015; Boavida et al., 2016).

Industry 4.0 and Context Interoperability in IoT

When considering IoT usage in industrial environments, the term Industrial Internet of Things (IIoT), or simply Industrial Internet, is employed, and is being considered as fundamentally linked to Industrie 4.0 (I4.0) (Jeschke et al., 2017). Comprising technologies such as IIoT, robotics/automation/control, additive manufacturing, simulation, cloud-based computing and platforms, industrial security, cognitive computing and artificial intelligence, mobility and wearables, big data and analytics, systems integration, augmented and virtual reality, as well as smart and new materials, I4.0 gives rise to new services and business models (Frank et al., 2019), driven by such technologies. Product Lifecycle Management (PLM) systems are particularly benefitting from such technologies to connect physical assets and products, processes, data, people and business systems (Keivanpour and Ait Kadi, 2019) exploiting product embedded sensor and intelligence capabilities, including product or process Condition Monitoring (CM) capabilities. Recent developments in IoT technologies have led to a renewed interest in context-aware computing. Context-awareness plays a central role in defining what data needs to be collected and how to be processed, as well as in determining what information and services are required to be made available to a consumer of data or services (Perera et al., 2014; Sezer et al., 2018). Context management is considered to comprise context acquisition, modeling, reasoning, and dissemination (Perera et al., 2014). **Table 2** summarizes surveys of IoT context-aware systems from 2010 to 2020.

Context information can be provided in various different ways, including variations in format, length, type, and representation of the data (de Matos et al., 2020). Hence, there is a need to ensure that context sharing platforms offer context interoperability. Context-relevant data can be produced by IoT entities and context management needs to be handled through a context management information processing layer. This layer would be expected to handle context data produced from multiple sources, including third-party software. Therefore, context sharing functionality is facilitated by a context sharing platform. The platform is capable of creating semantic interconnections between domains *via* the sharing of context information. As IoT environments can be highly complex, context-sharing platforms must be capable of dealing with a range of situations and implement service adaptation mechanisms driven by context building blocks (de Matos et al., 2020). These building blocks can be categorized as Properties and Architectural Components. The former applies to predominantly software aspects of context sharing platforms, including Modeling, Reasoning, Dissemination, Processing, Interoperability, Privacy, Scalability, and Availability, as shown in **Table 3**.

Architectural considerations regarding enabling hardware for the deployment of a context sharing platform, include communication technologies, storage space, and processing layers. Furthermore, some building blocks are strictly related to the context sharing properties (e.g., Modeling, Reasoning, and Dissemination), which are those that are specifically required in industrial monitoring. There are a variety of different IoT platforms, frameworks, services and middleware that are capable of collecting, processing and analyzing sensor data. In this regard, various researchers (Perera et al., 2015; Mineraud et al., 2016; Sezer et al., 2018; Pradeep and Krishnamoorthy, 2019; de Matos et al., 2020) have produced surveys of such IoT platforms, frameworks, systems, prototypes, middleware, and various different techniques and some representative examples are listed in **Table 4**, showing also their context modeling, reasoning, and dissemination features.

While all the approaches deal with some form of context management, starting from acquisition and modeling, eventually actionable context needs to be domain—specific. In the application domain of asset and maintenance management, context strongly depends on assets and their hierarchy. Unless such context is captured, it is hard to convert IoT-generated data from industrial systems to actions. Therefore, it is important to create a representation that integrates qualitative and quantitative

TABLE 2 | Summary of IoT context-aware system surveys.

Survey title	Year	Contribution	References
A survey of context modeling and reasoning techniques	2010	State-of-the-art in context modeling and reasoning in pervasive computing.	Bettini et al., 2010
Context aware computing for the internet of things: a survey	2014	Comprehensive survey and analysis of context awareness for internet of things.	Perera et al., 2014
Engineering context-aware systems and applications: a survey	2016	Context-aware systems and applications in engineering.	Alegre et al., 2016
Internet of things: a review of surveys based on context aware intelligent services	2016	A meta-survey of surveys on context awareness	Gil et al., 2016
Context-aware computing, learning and big data in internet of things: a survey	2018	Context awareness for IoT	Sezer et al., 2018
The MOM of context-aware systems: a survey	2019	Comparison of several context-aware systems	Pradeep and Krishnamoorthy, 2019
Context information sharing for the internet of things: a survey	2020	Presented essential building blocks for the development of context sharing platforms and reviewed the challenges and open issues for such platforms.	de Matos et al., 2020

data, wherein data and service delivery is determined upon resolving the apparent context of a service or data request. The most common approaches to achieve this, as seen in **Table 3**, is through ontology-based modeling. An ontology formally represents knowledge through concepts and relationships that exist in a specific domain and are a key construct of the semantic web (Gayathri and Uma, 2018).

Ontologies in Predictive Maintenance and Asset Management

As the manufacturing environment is becoming knowledge-intensive and more dynamic, maintenance is becoming more and more crucial in Asset Lifecycle Management. The use of semantic technologies, particularly ontology-based modeling for predictive maintenance, has become an important research topic and thus many ontologies have been offered to promote knowledge representation and reuse within the context of predictive maintenance. Medina-Oliva et al. (2014) developed a knowledge model for fleet predictive maintenance to handle fleet-wide contextual knowledge, arguing that fleet-wide Prognostics and Health Management (PHM) requires a knowledge-based system capable of handling contextual information. Thus, decision-making processes for diagnosis and maintenance are strengthened by semantic modeling, which deals the definition of concepts and relationships between them. In another example, an ontology was developed for the predictive maintenance in the wind energy domain and used as a basis for the identification and diagnosis of faults for monitoring the condition of wind turbines (Papadopoulos and Cipcigan, 2010). The proposed ontology model was used, by conducting ontology queries, to detect potential failures and their specific locations in the gearbox of the Wind Energy Converter (WEC).

When considering the manufacturing domain, it is of interest to capture the functional impact of asset integrity level on the actual manufacturing process. Castet et al. (2018) presented an approach for capturing fault information in a modeling environment using ontology of fault management and a set of plugins designed to automatically extract two reliability artifacts, the FMECA and fault tree. FMECA offers a sound basis upon which to express the organizational and functional association between a manufacturing asset hierarchy and its linkage with the functional integrity of the production facility. In the same year, Nuñez and Borsato (2018) conducted another study proposing an ontological model called OntoProg, serving as a widely agreed data and knowledge representation scheme for diagnostic-oriented maintenance, capable of being used in different types of industrial machines, and a set of SWRL rules to improve the ontology's expressiveness were suggested. In another recent example of an ontology-based approach to predictive maintenance, fuzzy clustering is employed to infer the criticality of failures, while SWRL rules are employed for predictive reasoning for the transition between states of different criticality, without though applying context-specific modeling an reasoning (Cao et al., 2019). Ontological approaches to support maintenance management that employ industrial scenarios have been developed for a range of assets, including urban infrastructure (Wei et al., 2020), highway infrastructure (France-Mensah and O'Brien, 2019), Building Information Management (BIM) (Farghaly et al., 2019), transport infrastructure (Ren et al., 2019; Li et al., 2020), and railway infrastructure (Dimitrova et al., 2020). **Table 5** summarizes of ontologies in maintenance and asset management.

The review of the related research work reveals two issues. Firstly, there is a missing link between knowledge constructs and context-dependent operational reliability-based services adaptation actions. Focusing on the asset context, relevant domain knowledge can be modeled in many forms but of particular interest are knowledge constructs relevant to reliability analysis, such as Fault Modes, Effects (and criticality) Analysis, FME(C)A. FMEA or FMECA models are however often utilized as a design-stage engineering study. Maintenance services, on the other hand, need to be invoked during the operating time and, thus, relevant information representations need to be enriched to enable dynamic context

TABLE 3 | Context sharing concerns.

Context sharing properties	Type	Aim	Implementation features	References
Modeling (M)	Properties	Responsible for mapping context into a predefined format.	Key-value, markup scheme, graphical, object oriented, logic-based, ontology-based, and hybrid context modeling.	Chen and Kotz, 2000; Perera et al., 2015
Reasoning (R)	Properties	Defined as the process to obtain high-level information from less enriched, or even raw data	Supervised and unsupervised learning, rules, fuzzy logic, ontology-based, probabilistic reasoning.	Bettini et al., 2010; Perera et al., 2015
Data Dissemination (D)	Properties	The context information is shared to relevant entities	Static and dynamic.	Perera et al., 2015
Privacy (P)	Properties	Data on the context includes private data, such as User ID, preferences, activities, and location. Although these drive context, privacy preservation should apply.	Access control policies, anonymization, cryptography.	Tiburski et al., 2015
Interoperability (I)	Properties	Heterogeneity of data requires that different subsystems or systems must be interoperable	Interoperability through format, source, length, and representation, and semantic alignment	de Matos et al., 2020
Context Processing (CP)	Properties	It aims to obtain, produce, and share context information to service a data or service request	Searching, filtering, and aggregation.	Lunardi et al. (2015)
Availability (AV)	Properties	The context must be always available for possible sharing	Availability ensured *via* cloud platforms or cached data	de Matos et al., 2020
Communication technologies (C)	Architectural Components	It refers to all equipment and programs that are used to process and communicate information	Communication devices, channels, and protocols for external and internal networks	Doukas et al., 2015; de Matos et al., 2020
History (Hi)	Architectural Components	Past data or inferred context stored locally or over the cloud.	Locally or cloud—based	de Matos et al., 2020
Architectural model	Architectural Components	Architecture can follow different patterns to support context sharing	Cloud-based, centralized-edge, and decentralized-edge	de Matos et al., 2020

inference and the composition of contextually relevant services. Secondly, existing predictive maintenance approaches in the manufacturing domain are still limited to the deployment of condition monitoring systems for identifying the failure mode and effects analysis in mechanical components. Therefore, the resolution of asset context is needed to analyse mechanical systems and logically connect measurements, observed behavior and intended function, with machinery operating condition and faults. To this end, FMEA offers appropriate grounding for the baseline of the knowledge mapping. According to Keivanpour and Ait Kadi (2019), failure mode analysis based on FME(C)A is recommended to ensure that maintenance activities are consistent with established fundamental practice-oriented knowledge. The following section presents an ontology-based development to describe knowledge through concepts and relationships that exist in a specific domain.

SYSTEM FRAMEWORK AND METHODOLOGY

This section presents the design of a system framework to develop a maintenance context ontology focused on failure analysis of mechanical components so as to drive monitoring services adaptation. The proposed ontology for the context resolution mechanism is relevant to failure analysis of mechanical components, and the terminology and relationships between concepts are structured on the basis of relevant standards with a reliability-oriented knowledge grounding. A mechanism for reasoning is being utilized for the delivery of context resolution and the obtained context can introduce a metadata layer on data or events produced by either automation or human-driven means. An example of 6health management of rotating machinery is utilized to offer a basis for the domain context, but the actual upper level ontology expressiveness is such that can apply to a range of machines by extending it through more specialized or application specific detailed ontologies.

The ontology is being utilized for the storage of knowledge relevant to fault diagnosis and reliability analysis through monitoring techniques. Hence, it is possible to query which type of approach for condition monitoring should be used and in what manner. Thus, queries can be made about what kind of condition monitoring technique that should be used and how. Additionally, inferences can be drawn in the

TABLE 4 | Comparison of context-awareness features of existing approaches.

Approach name	Year	Category	Modeling	Reasoning	Dissemination	References
Context Toolkit	2001	Toolkit	Key-value	(✓) Provided but not mentioned	Query	Dey et al., 2001
Aura	2002	Middleware	Markup Schemes	Rules	Publish	Garlan et al., 2002
CoBrA	2004	Middleware	Ontology-Based	Rules, ontology-based	Query	Chen et al., 2004
CARS	2005	System	Key-Value	Un-Supervised	(✓) Provided but not mentioned	Wilson et al., 2005
MoCA	2007	Middleware	Markup Schemes, Ontology-Based	Ontology-Based	Publish, Query	de Rocha and Endler, 2006
CoSM	2009	Model	Ontology-Based	Ontology-Based	Dynamic	Yamamoto et al., 2009
ConCon	2014	Middleware	Key-Value	Ontology-Based	Static	Madhukalya and Kumar, 2014
RCOS	2016	Middleware	Ontology-Based	Ontology-Based	Dynamic	Dhallenne et al., 2016
PSW	2017	Model	Ontology-Based	Ontology-Based, Rules	Dynamic	Ruta et al., 2017
CoaaS	2018	Middleware	(✓) Provided but not mentioned	Rules, Pro	Dynamic	Hassani et al., 2018
SCENTS	2019	Middleware	(✓) Provided but not mentioned	Rules	Dynamic	Liu et al., 2019

TABLE 5 | Summary of ontologies in maintenance and asset management.

Survey Title	Ontology domain	Contribution	References
A formal ontology for semantics in maintenance platforms	Production system	An ontology to produce new knowledge in the field of industrial maintenance that supports decision-making in the maintenance process.	Karray et al., 2012
Ontology-based implementation of an advanced method for time treatment in asset lifecycle management	Lathe machine	Implemented a method for exploiting the characteristics of time in maintenance asset lifecycle management (ALM) systems.	Matsokis and Kiritsis, 2012
Ontology-based schema to support maintenance knowledge representation with a case study of a pneumatic valve	Pneumatic valve	A methodology for knowledge representation using ontology concepts is proposed to overcome the problems of heterogeneity and inconsistency in maintenance records.	Ebrahimipour and Yacout, 2015
A novel maintenance system for equipment serviceability improvement	Manufacturing machine	A maintenance system for real-time equipment that integrates augmented reality (AR) for context-aware overlay of textual and graphical maintenance instructions on the maintenance scene.	Ong and Zhu, 2013
Context-aware recommendation for industrial alarm system	A power generation plant	An industrial alarm management system through semantic web technology and machine learning techniques.	da Silva et al., 2018
Semantic data model for operation and maintenance of the engineering asset	N/A	Proposed a semantic data model for engineering asset management, focusing on the operation and maintenance phase of its life cycle.	Koukias et al., 2013
Context modeling with situation rules for industrial maintenance	knowledge gateway system	A knowledge modeling approach and a technical architecture of a gateway system developed to support maintenance personnel.	Aarnio et al., 2016
A research on intelligent fault diagnosis of wind turbines based on ontology and FMECA	Wind turbine	A method of intelligent wind turbine fault diagnosis based on ontology and FMECA is proposed.	Zhou et al., 2015
Building an ontological knowledgebase for bridge maintenance	Transport infrastructure	"An ontology to achieve automatic rule checking and improve the management and communication of knowledge related to bridge maintenance."	Ren et al., 2019

sense that it is possible to make a comparison between an obtained value and specific thresholds based on relevant ISO standards in order to determine whether the value can be categorized as Good, Satisfactory, Alert or Alarm. Therefore, if the recorded value is considered to be in the Alert category, the system diagnoses that a failure could occur and a maintenance notification is issued for the machine indicating that intervention is required. Subsequent to the identification of an alert notification, it is then necessary to connect it with diagnostic information of the mechanical part being investigated, which will allow the failure mode and the potential causes to be determined.

Nonetheless, such simple threshold-based rules often fail to apply in practice and in view of that the ontological

approach does not seek to replace actual diagnostics techniques, which may involve far more efficient and sophisticated data processing. Instead it acts as a meta-layer of knowledge to drive services adaptation, and as such could work in synergy with other techniques of data processing and condition monitoring approaches. The intended end result is that the proposed maintenance intervention is more directed, and tailored to the apparent context of a situation. The process was applied as follows:

- A concept knowledge base is established and created on the basis of the professional expertise of mechanical engineers. The fundamental concept knowledge pertaining to the domain of condition monitoring is founded on professional expertise, associated studies and standard specifications. Extraction and representation of the necessary signal functions is then performed.
- The knowledge is then transformed into ontology and SWRL rules. The ontology editor Protégé 5.5 along with its plugins (e.g., SWRL editor) are utilized in this stage.
- Implicit knowledge is extracted from the knowledge base by the ontology management system according to the SWRL rule engine.
- Identification of the source of the vibration is performed using the obtained signal as input and then conducting Pellet reasoning. **Figure 1** provides more details about the system.

Maintenance Ontology Development

The development of ontology can be based on one of the numerous procedures described in the literature, including Uschold and King, Grüninger and Fox, Methodology, Ontology Development 101 (Noy and McGuinness, 2001) (OD1) and KACTUS. The OD1 is utilized in this study as it is broadly employed (Gong and Janssen, 2013; Lau et al., 2014; Nuñez and Borsato, 2018; Ren et al., 2019), has been demonstrated to be highly appropriate for maintenance modeling, and has been extensively documented for application in the Protégé environment.

The OD1 initial step is to determine the scope and domain. In this phase, the focus of the maintenance ontology is on modeling failure analysis of mechanical components to answer queries regarding how faults manifest themselves and how they can be prevented or addressed, so as to adapt relevant diagnostics or maintenance actions in a Condition-Based Maintenance setting. Therefore, the domain of the model is Maintenance.

The next OD1 steps are to consider reuse and enumerate terms. In this regard, ontological models developed by other researchers should be considered to determine their adaptability to the current research proposal such as those proposed in Nuñez and Borsato (2018), Sanislav and Miclea (2015). Moreover, this phase involves the enumeration of all terms pertinent to the area of the ontology being developed. Therefore, the main terminology and the associated definitions are based on consolidated academic literature and mostly on established international standards, such as condition monitoring, diagnostics and maintenance (ISO, 2012, 2017), vibration analysis (ISO, 2018), failure analysis (IEC 6030 0–1, IEC 6030 0-3-1), monitoring parameters: (ISO, 2011), asset management (ISO, 2014), and MIMOSA (www.mimosa.org) standards.

In the next phase, all classes and sub-classes are classified following a top-down approach. It starts with the most general definition of a domain concept and then continues with the more specialized concepts. In this context, the main class is of the Maintenance Ontology includes the subclasses Asset, FMEA_Technique and ConditionMonitoringParameters. Every such class has its own subclasses for example, subclass FMEA_Technique has subclasses: FailureEffect, FailureMode, PotentialCause, and Symptom. An example of class hierarchy is shown in **Figure 2A**.

The next steps in this process include the design of entities and properties. Entities are all the subjects of the studied domain; properties are verbs that clarify the relations between subjects and objects, or between-subjects themselves. Subsequent to defining the class hierarchy, it is important to determine the class relationships. They need to be accompanied by three distinct types of properties: object properties, data properties. The object attribute explains the associations among distinct classes. The data property explains the properties of certain occurrences both quantitatively and quantitatively. **Figures 2B,C** show the aforementioned object properties, data properties. The final stage is to create specific individual class instances within the class hierarchy, which involves: (1) selection of the class, (2) creation of an individual occurrence of the class, and (3) filling slot values. These instances are used in the representation of particular elements. **Figure 2D** presents class individuals.

Along with identification of the procedure that has been adopted, the development of ontology models requires tools that can support all activities in the development process. Such tools include TopBraid and OntoStudio, as well as open ones, such as the popular OntoEdit, HOZO and Protégé. Specifically, Protégé is the most dominant ontology publisher due to the fact that it is an open platform that offers plug-in extensibility as well as XML (S), OWL, RDF (S), and Excel support, along with graphic taxonomy, queries in SPARQL, rules in SWRL language, and a reasoner (Pellet). The combination of OWL/SWRL provides a more flexible ontology language for modeling knowledge domains with a greater degree of expressiveness than using OWL alone (Lawan and Rakib, 2019). The SWRL is a W3C recommendation that extends horn-clause rules to OWL. OWL has demonstrated significant expressive powers over other ontology languages as the recommended ontology language for the semantic web. While OWL ontologies provide simple, reusable and easy to understand domain knowledge models, they lack the declarative expressiveness offered by rules developed in SWRL.

IMPLEMENTATION ON A CASE STUDY

The applicability of the developed ontology model is shown by utilizing a real physical asset. Gearboxes have broad utilization in numerous applications such as machine tools, industrial devices, conveyors and essentially any form of rotatory power transmission equipment in which the torque

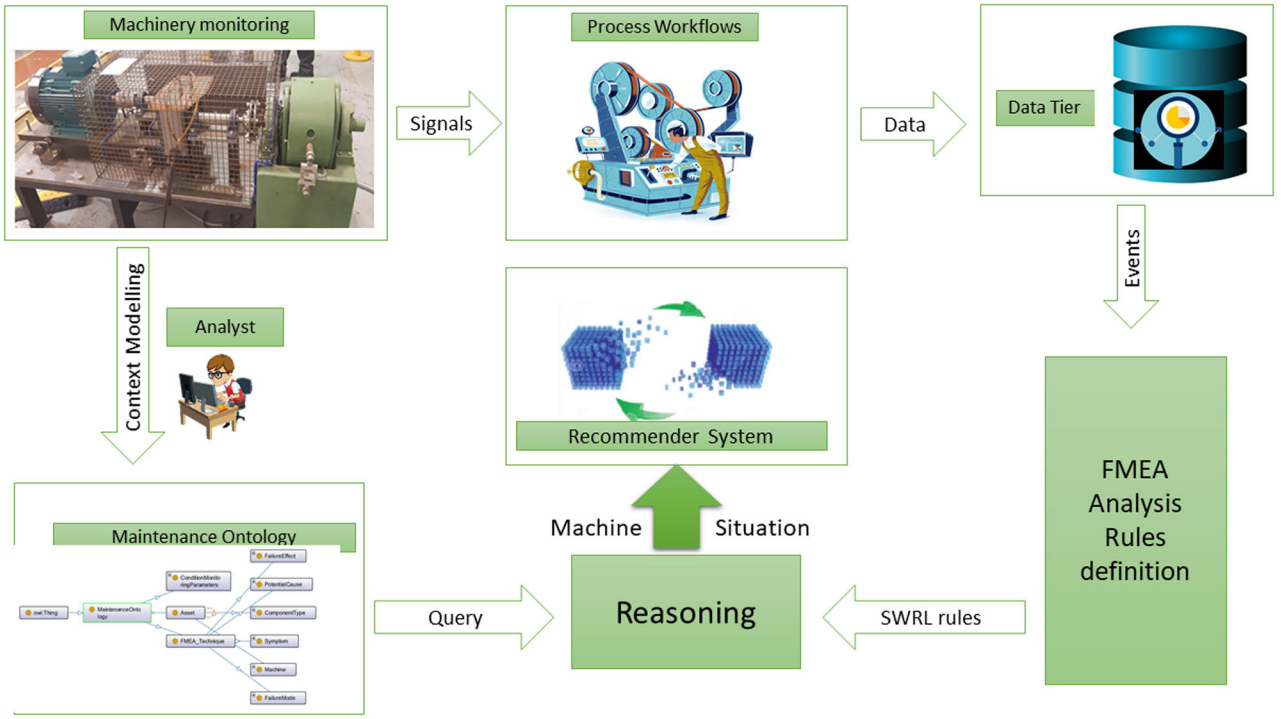


FIGURE 1 | Prototype framework.

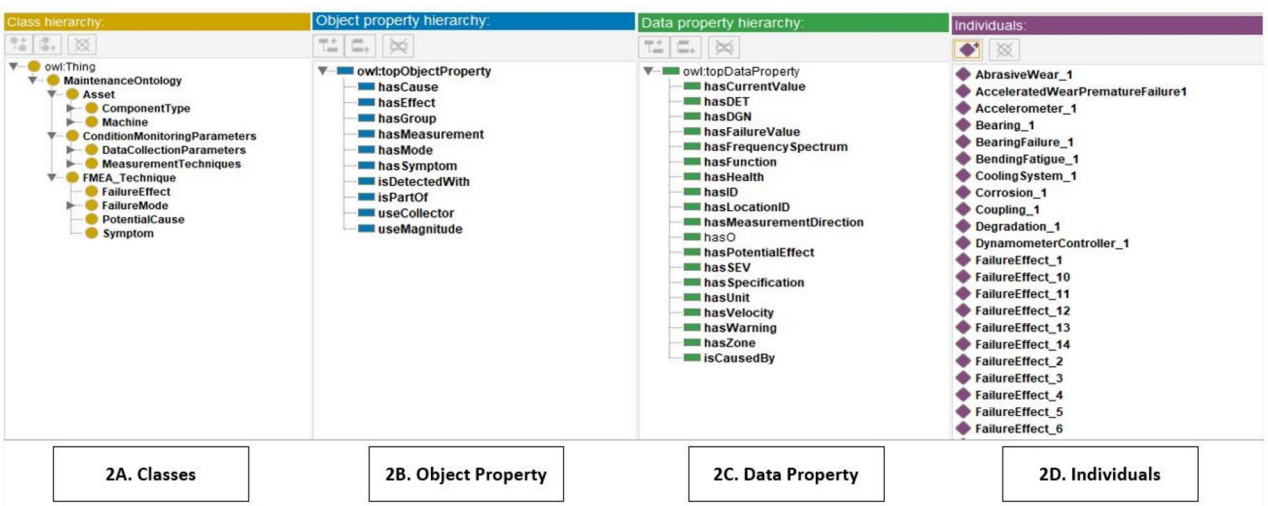


FIGURE 2 | Ontology classes, object property, data property, and individuals. **(A)** Classes. **(B)** Object property. **(C)** Data property. **(D)** Individuals.

and speed requirements need to be changed. If such devices fail, the results can often be catastrophic accidents with serious consequences. Therefore, a proactive approach must be adopted that enables such components to be monitored in real-time utilizing predictive maintenance methods (Khan et al., 2019). In the present study, the technique of vibration analysis is approached. Techniques used for assessing the health of components based on vibration are regarded as applicable for numerous reciprocating and rotating machines (Giurgiutiu et al., 2001; Bajrić et al., 2011). A laboratory-based test rig was employed and data was collected from its operation, and maintenance records. This has been designed for emulating

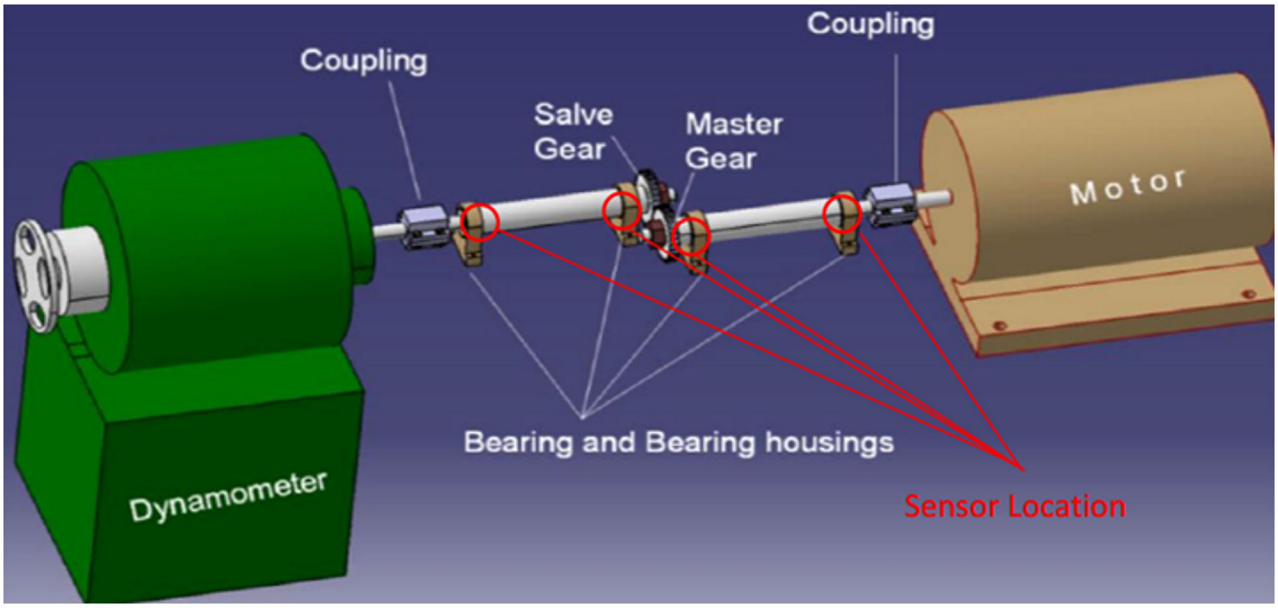


FIGURE 3 | CAD rendering of drive system and bearing locations.

complex cases of misalignment, relevant to manufacturing and aerospace engineering assets.

In order to capture the operational health of the machine, the test rig must be analyzed, allowing for an understanding on how to best capture the degradation effect on the test machine. As stated in ISO/FDIS 17359:2002(E), which details the flowchart for starting the condition monitoring process, the start is to choose the machine components in the maintenance ontology. Then, the necessary functionality of each of the components is explained for the machine to operate correctly. Additionally, all failure modes, effects, causes, symptoms and measurement approaches pertaining to the machine components are inputted utilizing the FMEA method. Subsequently, the implementation of the FMEA method indicates the most suitable measurement locations and their limits for the measurement of values by employing the vibration analysis method for prediction, which are based on ISO (2002), ISO (2016), and ISO (2009). The most pertinent components along with the most appropriate measurement methods are identified by utilizing the FMEA classification, which assigns weights according to the highest severity (SEV), occurrence (OCC), detectability (DET), and risk priority number (RPN).

Failure Mode and Effect Analysis

Asset context must be resolved for the analysis of mechanical systems and to establish logical connections between measurements, perceived behavior and the desired functionality, and the operating health and defects. In this regard, Fault Modes and Effects Analysis (FMEA) provides a suitable basis for the baseline of the knowledge mapping (IEC, 2018) due to various reasons. First, the qualitive components renders it suitable for the abstraction of maintenance knowledge focused on reliability. Second, the quantitative component allows maintenance tasks to be prioritized on the basis of measurements conducive to an approach based on risk. Third, its bottom-up structure allows failure to be assessed starting from the basic level of production systems; in other words, data are analyzed from machinery parts through to the overall system. The initial stage involves the determination of the specific aspects of the machine that have the potential to fail and then to comprehend the causes and effects of such failures (FMEA). Based on the study, it will become apparent where the data will be most accurate in highlighting the degradation of the machine being tested. Alongside the misalignment testing carried out on the machine, the rig can test the effects of loading through the dynamometer and the subsequent effects that loading will have on the system (**Figure 3**). Based on the FMEA **Table 6** (del Castillo et al., 2020), the most frequent outcome of misalignment of the gearbox will be vibration and power transfer loss through gearbox, as revealed by the RPN values. Subsequently, the vibration is spread across the machine and is most pronounced in specific locations, namely the bearings, and it is possible to easily capture the transfer loss by calculating the RPN difference between the driving motor and the loading dynamometer. As determined by the FMEA analysis, the two potential failures that are identified as having the greatest level of severity if misalignment or loading occur in the system are degradation of the gear teeth in the gearbox (RPN 150) as well as the bearing degradation (RPN 140).

Wearing of the teeth is generally caused by misaligned gears, excessive loading and lastly, a lack of lubrication. Degradation of the bearings is caused by wearing of the teeth in the gears as well as the impact of gear vibration being transferred to the shaft and then to the bearings. If the shaft of the gear is short

TABLE 6 | FMEA of test rig.

Item (ID)	Function (requirements)	Failure mode	Failure effects	SEV	Failure causes	OCC	Mitigation	DET	RPN
Bearing	"To achieve a smooth, low-friction rotary motion or sliding action between two surfaces"	Abrasive wear	Reduce fatigue life and misalignment in the bearing	6	lubricant condition, grease degradation, and improper isolation	4	Lubricant inspection and proper isolation, Monitor Shaft alignment	4	96
		Bearing seizure	Crack formation on rings and balls or rollers—Skidding	4	Inadequate heat removal capability—Loss of lubricant—High temperature—Excessive speed	3		3	36
		Noisy bearing	Surface fatigue—Glazing—Microspalling of stressed surfaces	4	Loss of lubricant—Housing bore out of round—Corrosive agents—Distorted bearing seals	3		2	24
		Fatigue (Spalling)	Bearing failure	3	Excessive loading (cyclic), misalignment	5		1	15
		Vibration	Scuffing—Fretting—Pitting of surfaces	7	Misalignment—Housing bore out of round—Unbalanced/excessive load	4		5	**140**
Gear	"To transmit shaft power on predetermined or designed angular velocities"	Tooth wear	Loss of rotation transfer, eventual gear vibration, noise	6	Contaminants in the gear mesh area or lubrication system	5	"Lubricant inspection, Regular inspection surface sanding"	5	**150**
		Scuffing	Wear and eventual tooth failure	5	Lubrication breakdown	2		4	40
		Tooth shear	Fracture	6	Tooth failure	2		3	36
		Spalling	Mating surface deterioration, welding, galling, eventual tooth failure	4	Fatigue	1		2	8
		Root filet cracking; Tooth end cracks	Surface contact fatigue and tooth failure	5	Tooth bending fatigue	2		2	20
		Pitting	Tooth surface damage	6	Cyclic contact stress transmitted through lubrication film	2		2	24

The bold values refer to components of high significance (RPN) in failure analysis.

and hard, and the bearings are situated in close proximity to the center where meshing of the gears occurs, this can be a source of vibration, which can be measured by placing sensors at the bearings as shown in **Figure 3**.

RESULTS AND DISCUSSION

The aim of ontology-based context modeling is to produce a semantic organization of data so as to drive maintenance services adaptation. When users interact with systems in this regard, the proposed maintenance ontology can help them (e.g., maintenance engineers) to narrow down the list of options by providing answers to questions such as:

- What are the common failures and diagnostic approaches for a given machine type?
- Which physical parameters to measure/use?
- What is the recommended preventive or corrective action for a specific failure mode of an asset?

An example of a typical utilization scenario is that during condition monitoring queries could be raised to resolve the monitoring service context. For instance, this could be related to determining the failure modes of a part, the functional effect of a defect on the operation of the test rig, the measurement alternatives suitable for specific defects and parts, in addition to the relevant measurement parameters. In the context of the present study, SPARQL queries were designed for the resolution of these queries. SPARQL additionally allows the federation of queries across different sources of data. By applying the following query, it is possible to determine "what are the main components of the Test rig?"

Figure 4 shows the results of a query to identify the main components of an asset type. These components are bearing,

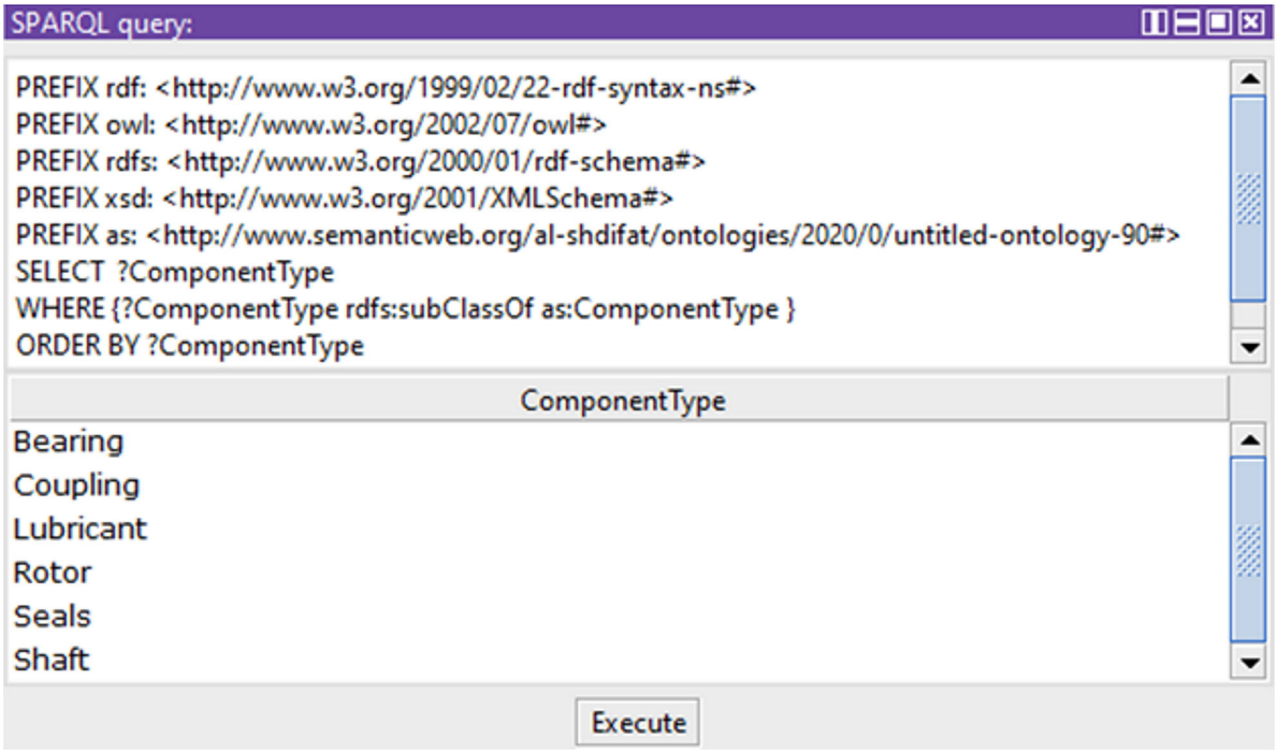


FIGURE 4 | Query result to identify the main components.

coupling, lubricant, rotor, seals, and shaft. Moreover, the present implementation allows a query in the maintenance ontology to resolve key analysis characteristics, such as components function, failure modes, causes, and effects. This query can be useful to a maintenance engineer in order to link faults to functional impacts, and other information to ensure the correct identification of the component being analyzed as shown in **Table 7**.

Another query can be applied based on FMEA to answer what are the common gearbox problems and diagnosis methods. For example, problems that arise in relation to gearboxes are misalignment, lubrication issues, bearing problems, gear teeth defects, thermal instability, and torsional and lateral vibration. Considering that the test rig under study was designed to study complex cases of misalignment in industrial machinery, the focus here will be on misalignment cases. Misalignment in gearbox arrangements can cause gear and bearing pitting, which eventually leads to complete failure. It may lead to vibrations and excessive loads that harm functioning components of the machine, such as bearings and oil seals. It is therefore important to detect and fix such issues to avoid incurring any unnecessary costs. As shown in **Figure 5**, this can take four forms: Axial misalignment; Offset or Parallel misalignment, when the centers of shafts are on different lines; Angular misalignment, when a motor shaft is at an angle to a driven component shaft; and Combination misalignment, when both angular misalignment and parallel misalignment occur.

After identifying the common gearbox problems, then it is important to identify parameters to be measured for fault detection. The developed ontology links physical measurement entities with appropriate measurement techniques. This allows to associate common faults with the physical asset and to match them with parameters or techniques appropriate for detecting the occurrence of such faults. For example, a component that has high significance in failure analysis is the bearing (**Table 8**). A critical failure mode is gear tooth wear and the typical failure effects for this is partial tooth contact (Misalignment). Another query can be applied to determine failure modes, failure causes, failure effects, symptoms, and fault severity (SEV), but also to determine the faults with highest diagnostic potential (DGN) or faults which pose the highest impact risk. SEV and DGN scale from 1 to 10, with the higher number representing the higher seriousness or risk and an appropriate query can return the faults with the highest DGN (**Table 8**) or risk. Therefore, parameters such as SEV, DGN and DET from the FMEA technique can be used within the ontology model to enable queries which in turn can identify components or processes of priority for maintenance actions.

Real data collection from the shop floor or simulated data (with "hasCurrentValue" data property) can be used to infer the component's health status and trigger alerts for decision making, such as the prognosis of a failure and the scheduling of Condition Based Maintenance (CBM) actions. In this regard, the SWRL language is being used in object properties to construct

TABLE 7 | Function, failure mode, failure cause, and failure effect for test rig.

Component	Function	Failure mode	Failure cause	Failure effect
"Shaft"	"It has the ability to translate in its axial direction, thereby changing the gear"	Misalignment1	"Angular misalignment of shaft due to mounting incorrect"	Abnormal temperature rise and excessive loading
		Corrosion	"Bearings exposed to corrosive environment"	Increased vibration and noise
"Lubricant"	"Lubricating the teeth and bearing / removing heat generated from operations"	Lubricant Degradation1	"Loss of lubricant, contaminated lubricant, aged lubricant, lubricant system failure, blocked lubrication filters, leakage"	Components failure and environmental pollution
"Bearing"	"To achieve a smooth, low-friction rotary motion or sliding action between two surfaces"	Fatigue	"Fatigue in rolling bearing parts by housing misalignment"	bearing failure
		Wear_1	"Lubricant condition, grease degradation, and improper isolation"	Sound_1
		Vibration	"misalignment"	Crack propagation
"Motor"	"Motors convert electrical energy into mechanical energy"	Overheating	"Cooling system failure, Temperature above limit, Temperature sensor failure."	UnableOperateMachine_1
		Shaft failure	"Overloading, fatigue, misalignment"	Halt generator operation and Increased vibration
		Bearing failure	"Bearing fatigue, Improper lubrication, lubricant contamination from dirt, abnormal vibration."	Increased vibration and noise

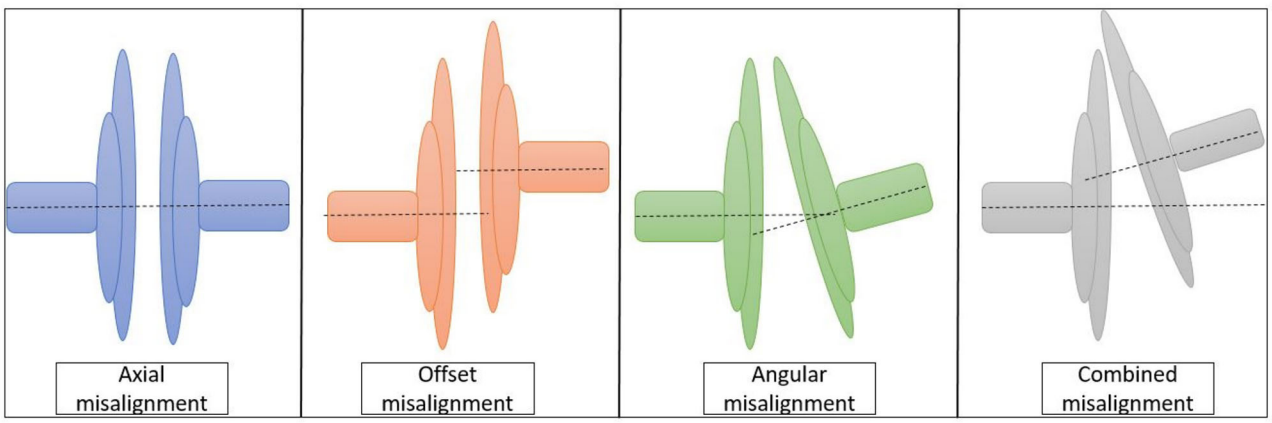


FIGURE 5 | Types of shaft misalignment adapted from Khan et al. (2019).

transitive rules (Nuñez and Borsato, 2018) and new connections are applied to the classes that allow assertion inferences to be improved. In this way the ontological approach becomes scalable: specifically SWRL built-ins (SWRLb) allow further extensions within a taxonomy. This greatly enhances the model by allowing multiple arguments according to specific real-world requirements, enabling greater expressiveness of OWL 2 languages. A transitive property is considered in cases such as: if subclass Component Type (C1) has object property has Mode, and subclass Failure Mode (FM) has object property has Cause (CA) related to subclass Potential Cause (PC), then subclass Component Type (C1) has the object property has Cause (CA) related to subclass Potential Cause (PC). Then the SWRL rule is: has Mode (?C1, ?FM1) Failure Mode (?FM1) Component (C1) Potential Cause (?PC1) has Cause (?FM1, ?PC1)—> has Cause (?C1, ?PC1). As part of the SWRL rules within the suggested ontology, (ISO, 2009) is utilized for the evaluation of the data gathered from the vibration measurements and analysis. The test rig employed in the pilot example is regarded as a mid-level asset in an asset hierarchy that includes a rolling-element bearing

TABLE 8 | Query outcome for failure mode with highest DGN.

Component type	Failure mode	Failure effect	Failure cause	Symptom	SEV	DGN	Technique
Rolling bearing	Tooth wear	Partial tooth contact (Misalignment)	Tooth failure	Vibration 3	4	5	Vibration analysis

(lower level in asset hierarch) that includes an accelerometer acting as a transducer in the data collection process. The resulting assessed parameters can be for example the velocity of the vibration in RMS mm/s, with the measurement sites as defined by standard MIMOSA VB-00, while the operating zone limits based on the (ISO, 2009) standard. To determine the machine's health status and recommended actions, the following main SWRL rules are set:

- **Component Health** (?CH1, ?E), equal(?E, "**Good**"), Measurement Location(?M1) -> has Warning(?CH1, "**Collect new data in 1 month**").
- **Component Health** (?CH 1, ?F), Measurement Location(?M1), equal(?F, "**Satisfactory**") -> has Warning(?CH1, "**Collect new data in 2 weeks**").
- **Component Health** (?CH 1, ?H), Measurement Location(?M1), equal(?H, "**Alert**") -> has Warning(?CH1, "**Schedule Condition-based Maintenance**").
- **Component Health** (?CH 1, ?I), Measurement Location(?M1), equal(?I, "**Alarm**") -> has Warning(?CH1, "**Turn off the machine**").
- **Has Health** (?CH1, ?E), equal(?E, "**Good**"), Measurement Location(?M1) -> **is Caused By** (?CH1, "**No Misalignment** ").
- **Has Health** (?CH1, ?F), Measurement Location(?M1), equal(?F, "**Satisfactory**") -> **is Caused By** (?CH1, "**Loss of lubricant - Housing bore out of round - Corrosive agents—Distorted bearing seals**").
- **Has Health** (?CH1, ?H), Measurement Location(?M1), equal(?H, "**Alert**") -> **is Caused By** (?CH1, "**Excessive loading (cyclic), misalignment** ").
- **Has Health** (?CH1, ?I), Measurement Location(?M1), equal(?I, "**Alarm**") -> **is Caused By** (?CH1, "**Misalignment—Housing bore out of round—Unbalanced/excessive load**").

Let's assume that when the data for the rolling bearing part exhibits an RMS mm/s value between zero and $\leq$2.3, then it should display a "good" notification. When values that exceed 2.3 but are below 4.5 are detected, it should display a "satisfactory" notification; a value between 4.5 and 7.1 should trigger an "alert" notification, and values in excess of 7.1 should cause an "alarm" notification that will instantly terminate the machine operation. Given this, let's assume that a value of 4.7 mm/s RMS is recorded. This is fed through the ontology, activating the Pellet plugin reasoner in the Protégé ontology editor. The outcome will be that the component's health will be inferred to be set as "Alert," producing a recommendation to "Schedule Condition-based Maintenance. Moreover, once an ALERT warning has been issued, it is then important to associate it with the diagnostic information of the analyzed mechanical component, associating the identified the failure mode with potential causes (**Figure 6**). In this way, the maintenance intervention becomes context-depended and is therefore more focused and relevant to the identified context of the monitoring situation.

While simple single-parameter threshold-based rules might be easy to interpret, they do not often hold in practice. Instead, more complex multi-parameter rules are more likely to apply. The reasoning process can replace simple rules with the activation of more complex decision functions which may be produced as a result of machine learning over monitoring historical data. The value of the described process is that it sits at a higher level of abstraction and can therefore work with different lower level computational rules.

CONCLUSION AND FURTHER RESEARCH

This paper presented the development of a context resolution service mechanism for industrial diagnostics, based on the design of a maintenance ontology focused on modeling and reasoning failure analysis of mechanical components. The maintenance ontology has been developed employing established methodologies and upon consulting a range of domain-relevant international standards. The ontology development was further applied on a physical mechanical transmission test rig. Thus, queries could be raised in terms of the resolution of the monitoring service context to determining the failure mode and its potential causes of the test rig, in addition to the relevant measurement parameters. Moreover, SWRL reasoning rules were used based on (ISO, 2009) for the evaluation of the data gathered, the prognosis of failure is being performed, sending a maintenance message for intervention in the machine. In this way, the maintenance intervention is more directed, ceasing to be exploratory. This highlighted the need for handling the whole context information management lifecycle and ontologies in maintenance and asset management to maximize the value delivered by physical engineering assets.

The outcomes of the work can be used in other industrially relevant application scenarios to drive maintenance service adaptation. While the application focus is quite specific, the ontology abstraction level is actually such that it could also be implemented on other application cases, as it offers a sound baseline for further customization or extensions. When serving different application scenarios, the derived abstract model developed with the process described in section system framework and methodology still holds, as the employed terms and relationships were developed employed established standards. However, after going through a similar process for deriving the application-specific part of the ontology as described in section implementation on a case study and the developed queries, as described in section results and discussion, additional needs can be identified, which may require

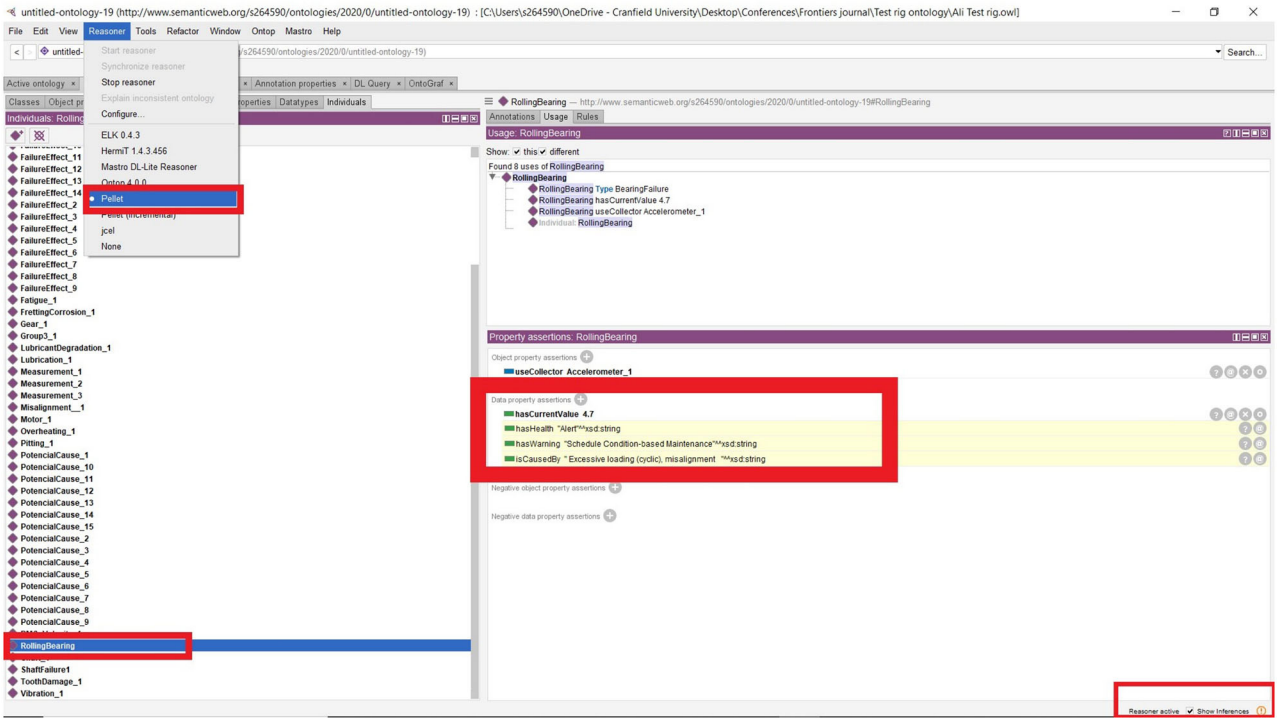


FIGURE 6 | SWRL rules for generating warning and potential causes.

the inclusion of additional entities, relationships, and queries development. This will be determined by going through an ontology assessment and evaluation cycle in the context of the new application scenario, especially regarding the ontology expressiveness and coverage. Nonetheless, this will affect largely lower abstraction level terms, rather than upper hierarchy classes and associated object or data properties. For example asset types and their associated terms if need be may be complemented by additional asset types. The higher level classes, object properties, and data properties will retain the structure of **Figures 2A–C** but the population of lower tier terms and individuals for such class structures will need to be developed for the additional asset types, as typically holds in managing ontologies. However, the example reasoning rules presented in Section results and discussion can be re-used but can be extended with additional ones to cover the coverage and expressiveness of the updated ontology.

Consequently, further research should be carried out to link the current ontology implementation with a live condition monitoring service, as well as to apply it to real industrial environments as an enabler of more efficient IoT-enabled monitoring services.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

AA-S and CE: original research, methodology, and paper editing. MK and AS: paper review and editing. All authors contributed to the article and approved the submitted version.

REFERENCES

Aarnio, P., Vyatkin, V., and Hastbacka, D. (2016). "Context modeling with situation rules for industrial maintenance," in *IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA).* (Berlin), 1–9. doi: 10.1109/ETFA.2016.7733539

Abowd, G. D., Dey, A. K., Brown, P. J., Davies, N., Smith, M., and Steggles, P. (1999). "Towards a better understanding of context and context-awareness," in *Proceeeding of 1st International Symposium on Handheld and Ubiquitous Computing, ser. HUC '99.* (*London, UK: Springer-Verlag),* 304–307. doi: 10.1007/3-540-48157-5_29

Abowd, G. D., and Mynatt, E. D. (2000). Charting past, present, and future research in ubiquitous computing. *ACM Trans. Comput. Hum. Interact.* 7, 29–58. doi: 10.1145/344949.344988

Alegre, U., Augusto, J. C., and Clark, T. (2016). Engineering context-aware systems and applications: a survey. *J. Syst. Softw.* 117, 55–83. doi: 10.1016/j.jss.2016.02.010

Al-shdifat, A., and Emmanouilidis, C. (2018). Development of a context-aware framework for the integration of internet of things and cloud computing for remote monitoring services. *Procedia Manuf. 16,* 31–38. doi: 10.1016/j.promfg.2018.10.155

Bajrić, R., Sprečić, D., and Zuber, N. (2011). Review of vibration signal processing techniques towards gear pairs damage identification. *Int. J. Eng. Technol.* 11, 124–128.

Bernardos, A. M., Tarrío, P., and Casar, J. R. (2008). "A data fusion framework for context-aware mobile services," in *Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, (South Korea: Seoul), 606–613. doi: 10.1109/MFI.2008.4648011

Bettini, C., Brdiczka, O., Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A., et al. (2010). A survey of context modelling and reasoning techniques. *Pervasive Mobile Comput.* 6, 161–180. doi: 10.1016/j.pmcj.2009.06.002

Boavida, F., Kliem, A., Renner, T., Jukka Riekki, J., Jouvray, C., Jacovi, M., et al. (2016). "People-centric internet of things–challenges, approach, and enabling technologies" in *Intelligent Distributed Computing IX—Symposium on Intelligent Distributed Computing (IDC'2015)* (Cham: Springer International Publishing), 463–474. doi: 10.1007/978-3-319-25017-5_44

Byun, H. E., and Cheverst, K. (2004) Utilizing context history to provide dynamic adaptations. *Appl. Artif. Intel.* 18, 533–548. doi: 10.1080/08839510490462894

Cao, Q., Samet, A., Zanni-Merk, C., De Beuvron, F. D. B., and Reich, C. (2019). An ontology-based approach for failure classification in predictive maintenance using fuzzy c-means and SWRL rules. *Procedia Comput. Sci.* 159, 630–639. doi: 10.1016/j.procs.2019.09.218

Castet, J. F., Bareh, M., Nunes, J., Okon, S., Garner, L., Chacko, E., et al. (2018). "Failure analysis and products in a model-based environment," in *IEEE Aerospace Conference Proceedings (IEEE)* (Big Sky, MT), 1–13. doi: 10.1109/AERO.2018.8396736

Chen, G., and Kotz, D. (2000). *A Survey of Context-Aware Mobile Computing Research.* Technical Report. Hanover, NH: Department of Computer Science, Dartmouth College, 1–16. Available online at: http://www.cs.dartmouth.edu/reports/TR2000-381.pdf (accessed June 16, 2020).

Chen, H., Finin, T., Joshi, A., Kagal, L., Perich, F., and Chakraborty, D. (2004). Meet the semantic web in smart spaces. *IEEE Internet Comput.* 8, 69–79. doi: 10.1109/MIC.2004.66

Chong, S. K., McCauley, I., Loke, S. W., and Krishnaswamy, S. (2007). "Context-aware sensors and data muling," in *Context Awareness for Self-Managing Systems (Devices, Applications and Networks) Proceeding* (Berlin: VDE-Verlag), 103–117.

da Silva, M. J., Pereira, C. E., and Götz, M. (2018). Context-aware recommendation for industrial alarm system. *IFAC-PapersOnLine* 51, 229–234. doi: 10.1016/j.ifacol.2018.08.266

de Matos, E., Amaral, L. A., and Hessel, F. (2017). *Context-Aware Systems: Technologies and Challenges in Internet of Everything Environments.* Cham: Springer International Publishing, 1–25. doi: 10.1007/978-3-319-50758-3_1

de Matos, E., Tiburski, R. T., Moratelli, C. R., Johann Filho, S., Amaral, L. A., Ramachandran, G., et al. (2020). Context information sharing for the internet of things: a survey. *Comput. Netw.* 166:106988. doi: 10.1016/j.comnet.2019.106988

de Rocha, R. C. A., and Endler, M. (2006). "Middleware: context management in heterogeneous, evolving ubiquitous environments," in *IEEE Distributed Systems Online* (IEEE), 1–13. doi: 10.1109/MDSO.2006.28

del Castillo, A., Diebolt, P., Hazi, C., and Kreinin, M. L. I. W. (2020). *Industrial internet of things for monitoring services*. (MSc Group Project Report), School of Aerospace, Transport and Manufacturing, Cranfield University, 15–19.

Dey, A., Abowd, G., and Salber, D. (2001). A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Hum. Comput. Interact.* 16, 97–166. doi: 10.1207/S15327051HCI16234_02

Dhallenne, J., Jayaraman, P. P., and Zaslavsky, A. (2016). *RCOS: Real Time Context Sharing Across a Fleet of Smart Mobile Devices.* Cham: Springer International Publishing, 87–100. doi: 10.1007/978-3-319-46301-8_8

Dimitrova, V., Mehmood, M. O., Thakker, D., Sage-Vallier, B., Valdes, J., and Cohn, A. G. (2020). An ontological approach for pathology assessment and diagnosis of tunnels. *Eng. Appl. Artif. Intellig.* 90:103450. doi: 10.1016/j.engappai.2019.103450

Doukas, C., Capra, L., Antonelli, F., Jaupaj, E., Tamilin, A., and Carreras, I. (2015). "Providing generic support for IoT and M2M for mobile devices," in: *The 2015 IEEE RIVF International Conference on Computing Communication Technologies—Research, Innovation, and Vision for Future (RIVF)* (Can Tho: IEEE), 192–197. doi: 10.1109/RIVF.2015.7049898

Ebrahimipour, V., and Yacout, S. (2015). "Ontology-based schema to support maintenance knowledge representation with a case study of a pneumatic valve," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems* (IEEE), 702–712. doi: 10.1109/TSMC.2014.2383361

Emmanouilidis, C., Koutsiamanis, R. A., and Tasidou, A. (2013). Mobile guides: taxonomy of architectures, context awareness, technologies and applications. *J. Netw. Comput. Appl.* 36, 103–125. doi: 10.1016/j.jnca.2012.04.007

Emmanouilidis, C., Pistofidis, P., Bertoncelj, L., Katsouros, V., Fournaris, A., Koulamas, C., et al. (2019). Enabling the human in the loop: linked data and knowledge in industrial cyber-physical systems. *Annu. Rev. Control.* 47, 249–265. doi: 10.1016/j.arcontrol.2019.03.004

Farghaly, K., Abanda, F. H., Vidalakis, C., and Wood, G. (2019). BIM-linked data integration for asset management. *Built Environ. Project Asset Manage.* 9, 489–502. doi: 10.1108/BEPAM-11-2018-0136

France-Mensah, J., and O'Brien, W. J. (2019). A shared ontology for integrated highway planning. *Adv. Eng. Inform.* 41:100929. doi: 10.1016/j.aei.2019.100929

Frank, A. G., Dalenogare, L. S., and Ayala, N. F. (2019). Industry 4.0 technologies: implementation patterns in manufacturing companies. *Int. J. Prod. Econ.* 210, 15–26. doi: 10.1016/j.ijpe.2019.01.004

Garlan, D., Siewiorek, D. P., Smailagic, A., and Steenkiste, P. (2002). 'Project aura: toward distraction-free pervasive computing. *IEEE Pervasive Comput.* 1, 22–31. doi: 10.1109/MPRV.2002.1012334

Gayathri, R., and Uma, V. (2018). Ontology based knowledge representation technique, domain modeling languages and planners for robotic path planning: a survey. *ICT Express* 4, 69–74. doi: 10.1016/j.icte.2018.04.008

Gil, D., Ferrández, A., Mora-Mora, H., and Peral, J. (2016). Internet of things: a review of surveys based on context aware intelligent services. *Sensors* 16:1069. doi: 10.3390/s16071069

Giurgiutiu, V., Adrian, C., and Goodman, P. (2001). *Review of Vibration-Based Helicopters Health and Usage Monitoring Methods.* Columbia: *University of South* Carolina, Department of Mechanical Engineering.

Gong, Y., and Janssen, M. (2013). An interoperable architecture and principles for implementing strategy and policy in operational processes. *Comput. Ind.* 64, 912–924. doi: 10.1016/j.compind.2013.06.018

Hassani, A., Medvedev, A., Haghighi, P. D., Ling, S., Indrawan-Santiago, M., Zaslavsky, A., et al. (2018). "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* (Athens: IEEE), 385–390. doi: 10.1109/PERCOMW.2018.8480240

Henricksen, K. (2003). *A framework for context-aware pervasive computing applications*. (Ph. D thesis). The School of Information Technology and Electrical Engineering, The University of Queensland, 13–20.

IEC 60812 (2018). *Failure Modes and Effects Analysis (FMEA and FMECA)* (Geneva: IEC).

ISO 10816-3 (2009). *Mechanical Vibration—Evaluation of Machine Vibration by Measurements on Non-Rotating Parts* (Geneva: ISO).

ISO 13306 (2017). *Maintenance—Maintenance Terminology* (Geneva: ISO).

ISO 13372 (2012). *Condition Monitoring and Diagnostics of Machines—Vocabulary* (Geneva: ISO).

ISO 13373-1 (2002). *Condition Monitoring and Diagnostics of Machines—Vibration Condition Monitoring* (Geneva: ISO).

ISO 13373-2 (2016). *Condition Monitoring and Diagnostics of Machines—Vibration Condition Monitoring* (Geneva: ISO).

ISO 17359 (2011). *Condition Monitoring and Diagnostics of Machines*: *General Guidelines* (Geneva: ISO).

ISO 2041 (2018). *Mechanical Vibration, Shock and Condition Monitoring—Vocabulary* (Geneva: ISO).

ISO 55000 (2014). *Asset Management—Overview, Principles and Terminology* (Geneva: ISO).

Jeschke S., Brecher C., Meisen T., zdemir D., and Eschert T. (2017). "Industrial internet of things and cyber manufacturing systems," in *Industrial Internet of Things. Springer Series in Wireless Technology,* eds S. Jeschke, C. Brecher, H. Song, and D. Rawat (Cham: Springer). doi: 10.1007/978-3-319-42559-7_1

Karray, M. H., Chebel-Morello, B., and Zerhouni, N. (2012). A formal ontology for industrial maintenance. *Appl. Ontol.* 7, 269–310. doi: 10.3233/AO-2012-0112

Keivanpour, S., and Ait Kadi, D. (2019). Internet of things enabled real-time sustainable end-of-life product recovery. *IFAC PapersOnLine* 52, 796–801. doi: 10.1016/j.ifacol.2019.11.213

Khan, M. A., Shahid, M. A., Ahmed, S. A., Khan, S. Z., Khan, K. A., Ali, S. A., et al. (2019). Gear misalignment diagnosis using statistical features

of vibration and airborne sound spectrums. *Measurement* 145, 419–435. doi: 10.1016/j.measurement.2019.05.088

Koukias, A., Nadoveza, D., and Kiritsis, D. (2013). Semantic data model for operation and maintenance of the engineering asset. *IFIP Adv. Inform. Commun. Technol.* 398, 49–55. doi: 10.1007/978-3-642-40361-3_7

Lau, R., Li, C., and Liao, S. (2014). Social analytics: learning fuzzy product ontologies for aspect-oriented sentiment analysis. *Decis. Support Syst.* 65, 80–94. doi: 10.1016/j.dss.2014.05.005

Lawan, A., and Rakib, A. (2019). The semantic web rule language expressiveness extensions-a survey. *arXiv 1*1723.

Li, R., Mo, T., Yang, J., Jiang, S., Li, T., and Liu, Y. (2020). "Ontologies-based domain knowledge modeling and heterogeneous sensor data integration for bridge health monitoring systems," in *IEEE Transactions on Industrial Informatics* (IEEE), 3203. doi: 10.1109/TII.2020.2967561

Liu, C., Hua, J., and Julien, C. (2019). "SCENTS: collaborative sensing in proximity IoT networks," in *2019 IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops* (Kyoto: IEEE), 189–195. doi: 10.1109/PERCOMW.2019.8730863

Liu, W., Li, X., and Huang, D. (2011). "A survey on context awareness," in *2011 International Conference on Computer Science and Service System (CSSS)* (Nanjing: IEEE), 144–147. doi: 10.1109/CSSS.2011.5972040

Lunardi, W. T., de Matos, E., Tiburski, R., Amaral, L. A., Marczak, S., and Hessel, F. (2015). "Context-based search engine for industrial IoT: discovery, search, selection, and usage of devices," in *2015 IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA),* (Luxembourg), 1–8. doi: 10.1109/ETFA.2015.7301477

Madhukalya, M., and Kumar, M. (2014). "ConCon: context-aware middleware for content sharing in dynamic participating environments," in *Proceedings - IEEE International Conference on Mobile Data Management,* (Brisbane, QLD: IEEE), 1, 156–161. doi: 10.1109/MDM.2014.25

Matsokis, A., and Kiritsis, D. (2012). "Ontology-based implementation of an advanced method for time treatment in asset lifecycle management," *in: Engineering Asset Management and Infrastructure Sustainability. eds J. Mathew, L. Ma, A. Tan, M. Weijnen, J. Lee (London, UK: Springer),* 647–662. doi: 10.1007/978-0-85729-493-7_50

Medina-Oliva, G., Voisin, A., Monnin, M., and Leger, J. B. (2014). Predictive diagnosis based on a fleet-wide ontology approach. *Knowledge-Based Syst.* 68, 40–57. doi: 10.1016/j.knosys.2013.12.020

Mineraud, J., Mazhelis, O., Su, X., and Tarkoma, S. (2016). A gap analysis of Internet-of-Things platforms. *Comput. Commun.* 89–90, 5–16. doi: 10.1016/j.comcom.2016.03.015

Noy, N. F., and McGuinness, D. L. (2001). A guide to creating your first ontology. *Biomed. Inform. Res.* 7–25. Available online at: https://protege.stanford.edu/publications/ontology_development/ontology101.pdf (accessed June 15, 2020).

Nuñez, D. L., and Borsato, M. (2018). OntoProg: an ontology-based model for implementing prognostics health management in mechanical machines. *Adv. Eng. Inform.* 38, 746–759. doi: 10.1016/j.aei.2018.10.006

Ong, S. K., and Zhu, J. (2013). A novel maintenance system for equipment serviceability improvement. *CIRP Ann. Manuf. Technol.* 62, 39–42. doi: 10.1016/j.cirp.2013.03.091

Papadopoulos, P., and Cipcigan, L. (2010). "Wind turbines condition monitoring: an ontology model," in *International Conference on Sustainable Power Generation and Supply* (Nanjing: IEEE), 1–4. doi: 10.1109/SUPERGEN.2009.5430854

Perera, C., Liu, C. H., Jayawardena, S., and Chen, M. (2015). A survey on internet of things from industrial market perspective. *IEEE Access* 2, 1660–1679. doi: 10.1109/ACCESS.2015.2389854

Perera, C., Zaslavsky, A., Christen, P., and Georgakopoulos, D. (2014). Context aware computing for the internet of things. *IEEE Commun. Surveys Tutorials* 16, 414–454. doi: 10.1109/SURV.2013.042313.00197

Perttunen, M., Riekki, J., and Lassila, O. (2009). Context representation and reasoning in pervasive computing: a review. *Int. J. Multimedia Ubiquit. Eng.* 4, 1–28.

Pradeep, P., and Krishnamoorthy, S. (2019). The MOM of context-aware systems: a survey. *Comput. Commun.* 137, 44–69. doi: 10.1016/j.comcom.2019.02.002

Ramachandran, G. S., and Krishnamachari, B. (2019). Towards a large scale IoT through partnership, incentive, and services: a vision, architecture, and future directions. *Open J. Int. Things 5,* 80–92.

Ren, G., Ding, R., and Li, H. (2019). Building an ontological knowledgebase for bridge maintenance. *Adv. Eng. Softw.* 130, 24–40. doi: 10.1016/j.advengsoft.2019.02.001

Rizou, S., Haussermann, K., Durr, F., Cipriani, N., and Rothermel,. K. (2010). "A system for distributed context reasoning," in *Sixth International Conference on Autonomic and Autonomous Systems (ICAS)* (Cancun: IEEE), 84–89. doi: 10.1109/ICAS.2010.21

Ruta, M., Scioscia, F., Ieva, S., Loseto, G., Gramegna, F., Pinto, A., et al. (2017). "Knowledge discovery and sharing in the IoT: the physical semantic web vision," in *Proceedings of the ACM Symposium on Applied Computing* (Marrakech), 492–498. doi: 10.1145/3019612.3019701

Sanislav, T., and Miclea, L. (2015). A dependability modeling approach for cyber-physical systems. *J. Comput. Sci. Control Syst.* 8, 23–28.

Schilit, B. N., and Theimer, M. M. (1994). "Disseminating active map information to mobile hosts." *IEEE Netw. 8,* 22–32. doi: 10.1109/65.313011

Sezer, O. B., Dogdu, E., and Ozbayoglu, A. M. (2018). Context-aware computing, learning, and big data in internet of things: a survey. *IEEE Internet Things J.* 5, 1–27. doi: 10.1109/JIOT.2017.2773600

Snidaro, L., García, J., and Llinas, J. (2015). Context-based information fusion: a survey and discussion. *Inform. Fusion* 25, 16–31. doi: 10.1016/j.inffus.2015.01.002

Statistica report. (2020). *Number of Internet of Things (IoT) Connected Devices Worldwide in 2018, 2025 and 2030.* Available online at: https://www.statista.com/statistics/802690/worldwide-connected-devices-by-access-technology/ (accessed June 22, 2020).

Tiburski, R. T., Amaral, L. A., Matos, E. D., Hessel, F. (2015). The importance of a standard security architecture for SOA-based IoT middleware. *IEEE Commun. Mag.* 53, 20–26. doi: 10.1109/MCOM.2015.7355580

Valverde-Rebaza, J. C., Roche, M., Poncelet, P., and de Lopes, A. (2018). The role of location and social strength for friendship prediction in location-based social networks. *Inform. Process. Manage.* 54, 475–489. doi: 10.1016/j.ipm.2018.02.004

van Bunningen, A. H., Feng, L., Apers, P. M. G. (2005). "Context for ubiquitous data management," *in Proceedings of the International Workshop on Ubiquitous Data Management, (Washington, DC),* 17–24. doi: 10.1109/UDM.2005.7

Wei, L., Du, H., Mahesar, Q. ain., Al Ammari, K., Magee, D. R., Clarke, B., et al. (2020). *A decision support system for urban infrastructure inter-asset management employing domain ontologies and qualitative uncertainty-based reasoning. Expert Syst. Appl. 158:113461.* doi: 10.1016/j.eswa.2020.113461

Wilson, D. H., Long, A. C., and Atkeson, C. (2005). "A context-aware recognition survey for data collection using ubiquitous sensors in the home," in *Conference on Human Factors in Computing Systems—Proceedings* (Portland, OR), 1865–1868. doi: 10.1145/1056808.1057042

Xu, L. D., He, W., and Li, S. (2014). Internet of things in industries: a survey. *IEEE Trans. Ind. Inform.* 10, 2233–2243. doi: 10.1109/TII.2014.2300753

Yamamoto, J., Nakagawa, H., Nakayama, K., Tahara, Y., and Ohsuga, A. (2009). "A context sharing message broker architecture to enhance interoperability in changeable environments," in *3rd International Conference on Mobile Ubiquitous Computing, Systems, Services, and Technologies* (Sliema: IEEE), 31–39. doi: 10.1109/UBICOMM.2009.48

Zhou, A., Yu, D., and Zhang, W. (2015). A research on intelligent fault diagnosis of wind turbines based on ontology and FMECA. *Adv. Eng. Inform.* 29, 115–125. doi: 10.1016/j.aei.2014.10.001

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2020 Al-Shdifat, Emmanouilidis, Khan and Starr. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



https://dspace.org/

Ontology-based context modeling in physical asset integrity management

Al-Shdifat, Ali; Emmanouilidis, Christos; Khan, Muhammad; Starr, Andrew G.

2020-10-30
Frontiers Research
https://doi.org/10.3389/fcomp.2020.578673
Al-Shdifat A, Emmanouilidis C, Khan M, Starr AG. (2020) Ontology-based context modeling in physical asset integrity management. Frontiers in Computer Science, Volume 2, October 2020, Article number 578673

Downloaded from a DSpace repository. https://dspace.org/