https://doi.org/10.55972/spectrum.v27i1.455

ORIGINAL PAPER

Spectrum - The Journal of Operational Applications in Defense Areas

Journal homepage: https://spectrum.ita.br

Operational Analysis and Logistics Engineering: Logistics Engineering

Building an Integrated Simulation Model with Ontologies for Prescriptive Maintenance

Mariana Teixeira Rosalin da Silva[1] and Henrique Costa Marques[1,*]

[1]Instituto Tecnológico de Aeronáutica (ITA), São José dos Campos/SP – Brazil

*Corresponding Author: hmarques@ita.br

Keywords

Ontology
Prescriptive Maintenance
Discrete-Event Simulation
SPARQL / Semantic Queries
Maintenance 4.0

Article Information

Received 31 July 2026
Revised 25 August 2026
Accepted 02 September 2026
Available 09 October 2026

Abstract

This article provides studies to support the creation of prescriptive maintenance frameworks, addressing the difficulty caused by the lack of formalized data that hinders the application of rule sets and the execution of data-driven maintenance measures. The absence of adequate data structuring makes it difficult to develop automated and efficient approaches to solve maintenance problems. One adopted solution is the use of semantic approaches through ontologies. By modeling the domain at a conceptual level, ontologies make it possible to establish relationships and decision logic to drive prescriptive maintenance solutions in an automated way, providing a coherent and consistent structure for the data and facilitating its understanding and manipulation. The use of ontologies enables a formal representation of the data, allowing the creation of more efficient frameworks for prescriptive maintenance. Using a solution that integrates a simulation model with an ontology simultaneously would bring significant gains to the academic community, since it is possible to obtain benefits in terms of in-depth data understanding, automation of prescriptive maintenance measures, and efficiency in creating frameworks to solve maintenance problems.

1. Introduction

The Industry 4.0 (I4.0) revolution brings great potential to increase the efficiency of maintenance activities through the automation of planning processes that are traditionally carried out by humans. However, prescriptive maintenance, despite its advantages, faces additional challenges when it comes to real-time replanning. This is due to unexpected system failures, the lack of sensors, the absence of a known degradation curve, or the need for a special treatment of a component when a failure occurs [9].

Prescriptive maintenance is an advanced form of preventive maintenance that involves prior knowledge of the causes of system degradation, of the tasks required for maintenance, and of the resources to be used [1], [2]. It means being able to identify the reason and the ideal moment, and how to have complete control of the system — optimizing processes, establishing indicators, and generating task lists and resource requests in advance — in order to avoid unplanned downtime until the ideal moment.

The capability to make use of decision-support system automation and to bring the planning specialist's knowledge into a computerized system is the task of interest of this article.

The peer review process was coordinated, and the final publication approved, by Humberto Baldessarini Pires.

The present work provides an experiment using ontologies of operations, RAMS (Reliability, Availability, Maintainability, and Safety), and maintenance, integrated into a discrete-event simulation model to collect relevant system data such as the mean time between failures, the mean time to repair, and the mean waiting time in the maintenance queue, among other performance indicators. These data can help evaluate the performance of the maintenance system and identify improvement opportunities, as well as support decision-making for maintenance task planning, being a fundamental piece for prescriptive maintenance to be developed.

Starting from the establishment of a case study in the aeronautical sector, ontologies are defined and semantic searches are performed on the knowledge base, showing the potential of establishing relationships between the concepts and the rules that guide the searches and allow extracting from the database the key elements that constitute a prescriptive maintenance plan.

This article is structured as follows. Section 2 describes the contextualization of the maintenance problem; Section 3 details the proposed solution and the use of ontologies with SPARQL, whose documentation is available in [20]; Section 4 presents the discussion of the results found and how they can be used in various contexts; and Section 5 provides the conclusion and future work.

2. Background on Maintenance Frameworks

Prescriptive maintenance frameworks are sets of guidelines, approaches, and methodologies used to implement predictive maintenance strategies, which aim to predict failures in equipment or systems before they occur and to take appropriate corrective actions.

These frameworks generally involve the following components:

- Monitoring — continuous collection of data and information about equipment performance, using sensors, remote monitoring systems, or other technologies;
- Data analysis — application of analytical techniques and advanced algorithms to extract relevant information from the collected data, identifying patterns, anomalies, and trends;
- Modeling and prediction — construction of predictive models based on the collected data, allowing the future behavior of the equipment to be forecast and possible imminent failures or problems to be identified;
- Diagnosis — interpretation of the results of the data analysis and predictive models to identify the root cause of potential failures and determine appropriate corrective actions; and
- Maintenance recommendations — based on the information obtained, prescriptive maintenance frameworks provide precise and specific recommendations on the maintenance measures to be taken, such as part replacement, repairs, or preventive interventions.

These computational frameworks are applied in various sectors — such as industry, manufacturing, energy, transportation, and healthcare — enabling the optimization of equipment availability, reliability, and efficiency, in addition to reducing operational costs and avoiding unplanned downtime.

Because each system has its own characteristics, there is no single algorithmic solution for prescription capable of solving all types of asset degradation. Therefore, a solution that is adaptable to various contexts must be developed. In this way, prescriptive maintenance frameworks need to be dynamic and adaptive; that is, it is necessary to build a solution for the modularization of these frameworks to make them holistic and capable of solving problems involving many types of demands for the different assets and resources available or to be allocated, using the same class of algorithms for different fleets and different complex systems.

Consequently, it is necessary to make these frameworks parameterizable so that they serve various contexts, whether an aircraft production line environment, commercial aircraft maintenance management, or even an automotive production line. Modularizing frameworks means ensuring broad applicability of the proposed approach to various industry contexts [3].

According to [10] and [11], this problem is already pointed out in the literature for both prescriptive maintenance frameworks whose application is restricted, reinforcing the need to create a model that allows them to become more generic in their applications.

3. Proposed Solution

Among the various ways of pursuing the dynamism of the frameworks that support prescriptive maintenance, this work seeks to achieve adaptation to various contexts based on their description through domain ontologies. The frameworks seek to define optimization algorithms that receive parameters from the assets and from the maintenance capacity that a given organization possesses. Since there are different levels of machine maturity, different operating models, and different maintenance structures, it is clear that such algorithms cannot adapt easily.

The proposed solution builds the context through ontologies and allows the algorithm to be used in more than one context by extracting the information and processing it so as to correctly send each parameter to the input interface. This work, however, will not go into the detail of an optimization algorithm, remaining within the scope of context description by making use of data capture through discrete-event simulation. In this way, a digital shadow will be proposed to elucidate how the ontology is used to generate the operating parameters of a scenario and is sent to a simulator that will exemplify the operation of various machines in different contexts, allowing a future optimization algorithm to make use of the generated events and to promote greater efficiency in the allocation of human and material resources in the scheduling of maintenance tasks.

3.1. Ontology for Context Definition (SPMOntology)

The use of ontologies is fundamental to obtaining optimized models, with rules, the mapping of the domain of the object of study, and the concepts of interest. In this case, the main problem faced by the developed frameworks is to provide the means to move from coexistence to interoperability and cooperation among different contexts within the same environment. Indeed, different researchers [12]–[15] agree that ontology engineering is recognized as the key technology to deal with this.

To define the objects, classes, attributes, and relationships of the ontologies, information was collected based on a Brazilian commercial aviation airline and on a company that manufactures military, commercial, and executive aircraft. Research and interviews were also conducted with specialists from the maintenance and operations areas. The RAMS ontology was based on the study of the CRIS (OSA-EAI) model specified by MIMOSA (Machinery Information Management Open Systems Alliance) [16].

In the literature we can find some applications of the use of ontologies to improve maintenance efficiency. The authors in reference [4] present an extensive literature review on semantic web and knowledge graph approaches for I4.0, addressing different perspectives such as standardizations, manufacturing, supply chain management, and predictive maintenance. The

generalized ontology model presented in [4] is based on the RAMI4.0 reference architecture for I4.0 and explores the concepts and relationships among manufacturing and maintenance areas such as date/time, location, sensing, equipment, products, processes, and facilities.

Also in the domain of industrial maintenance, in [5] a fault knowledge graph, SemKoRe, was developed. The concepts of Faults and Machine are modeled in the ontology to assist in maintenance and to enable fault resolution based on the fault history.

The work in this article differs from those mentioned by establishing the application to the domain of the Maintenance 4.0 hangar together with the events arising from fleet operation, such as unplanned and planned failures, in a simulation system for decision support. This section describes the three ontologies developed in the project: Operations, RAMS, and Maintenance, indicated in the framework work described in [10]. We call it the Smart Prescriptive Maintenance Ontology (SPMO).

Each ontology comprises classes, subclasses, attributes, and their relationships:

- Operations: classified with activity information, having Init Date, End Date, Activity Time, and Location Start and Location End;
- RAMS: classified with asset registration, condition, maintenance, and inter-system reliability information, having Agent, Segment, Asset, Health, Remaining Life, Event, Manufacturer, Model, Maintenance Plan, Work Request, and Work Order; and
- Maintenance: classified with information about the maintenance hangar/workshops, the maintenance team, and supplies, having Materials, Purchase Order, Inventory, Service Provider, and Man Power Shift, Specialty, and Man Power Available.

3.2. Implementing the Simulation Model

Using discrete-event simulation modeling, we intend to evaluate and optimize the performance of maintenance strategies, identify bottlenecks, estimate downtime, and assess the impact of different maintenance policies in a controlled environment. Therefore, this technique allows better maintenance management and helps maximize the availability and reliability of the assets [17], [18].

The essential components of the model were specified as shown in Table I. Discrete-event simulation modeling is suitable for representing dynamic systems where changes occur at discrete moments in time. In this case, each event would be a significant change in the system, such as the detection of a failure in a part, the start of scheduled maintenance, or the arrival of an aircraft at the maintenance queue. By using this approach, it is possible to model each component/asset of the aircraft, such as the individual parts, the systems, the subsystems, and the aircraft as a whole.

The amount of flight hours / activity time is simulated through the advancement of time in the simulation, considering the usage rate of the aircraft or machine. The detection of part failures is modeled based on probabilities or statistical distributions that represent the expected failure rate for each part/asset.

Table I. Input parameters for the simulation.

Entity	Attribute	Value (Example)
Operations	Flight Leg / Manufacturing Stages (Flights / Manufacturing)	40 daily flight legs / 7 processes per operation
	Number of Segments (Aircraft / Machines)	30
RAMS	Asset (Part / Equipment)	ENGINE CFM56 / Fooke CNC Milling Machine
	Task Scheduler (Failure Events)	Engine overload / Machine interruption
	Task Scheduler (Scheduled Maintenance Tasks)	Valve inspections / Cutting-tool inspection
Maintenance	Resource Availability (Slots / Manpower)	3

Scheduled maintenance stops are planned based on predetermined time intervals or on specific criteria, such as a minimum number of flight hours reached. The hangar maintenance queue is modeled as a data structure that records the aircraft waiting for service and the capacity order for performing maintenance, which can be distributed across more than one hangar according to the parameter provided to the simulator.

We can observe in Table I that the inputs can be parameterized for the different scenarios proposed in this work.

It is important to emphasize that the simulation modeling allows the model to be adapted to the specific characteristics of the asset maintenance system that one wishes to simulate.

3.3. Integration between the Simulation Model and the Ontologies with SPARQL (SPMOntology.API)

To integrate the ontologies with the simulator, it was necessary to create an API (Application Programming Interface) capable of exchanging information in real time. For the server, Apache Jena Fuseki [19] was chosen; it is a SPARQL server that allows RDF (Resource Description Framework) ontologies to be queried using the SPARQL query language [20]. The query is able to extract information such as instances, relationships, properties, and other items of interest. It provides a server environment to store and query RDF graphs, allowing the execution of queries, updates, insertions, and deletions in various ontologies through a previously configured dataset, with the architecture shown in Fig. 1.

3.4. The Simulator (SPMSimulator)

After specifying the simulation model, creating the ontological model, and defining the integration API methods, the components were incorporated into a simulation tool

developed in .NET C#, called SPMSimulator, so that it can receive the input parameters for the simulation of two case studies from different contexts:

- Line maintenance hangar; and
- Aircraft production line.

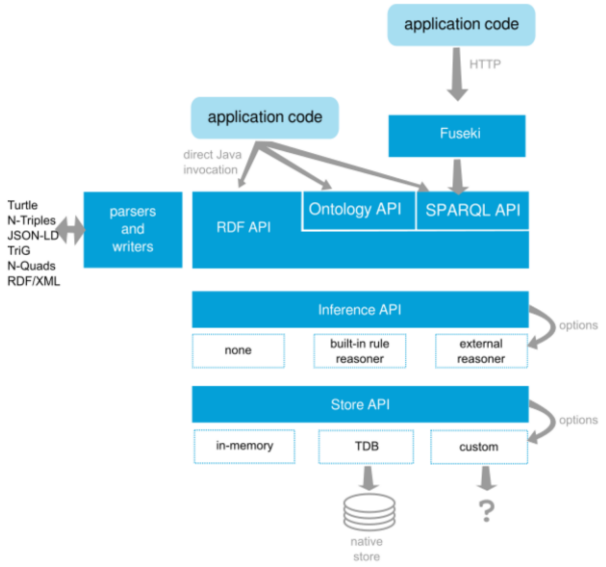


Fig. 1. Jena Architecture Overview. Source: [21].

The simulation tool was prepared to test the operation of the ontological structure through a reasoner (inference engine), applying the rules specified in the ontology (SPMO), and to process the results of failure time, repair time, and availability time of the assets of each segment (aircraft or machine) of the operation, changing in real time the current state of the segment according to the maintenance stops — whether due to failures and/or planned scheduling — and the waiting time in the queue. In this way, it is able to record events and provide inputs on the condition of the assets to support maintenance management. Fig. 2 shows the simulation of a flight operation scenario with 30 aircraft or segments simultaneously, which suffer random breakdowns characterizing the maintenance stops in a line maintenance hangar. At the center, the simulation model implemented for this scenario of interest is displayed, in which the entities move during the simulation; the tables at the bottom represent the data loaded from the ontologies, and the top part shows the simulator parameterization commands.

4. Results and Discussion

In this section we evaluate the potential of the ontologies created for the purpose of inferring knowledge about the main maintenance requirements that must be met by the maintainers for the aircraft fleet.

The data preparation and ontology analysis relied on the Protégé system [6] from Stanford University. For the queries, the SPARQL language [20] developed by the W3C (World Wide Web Consortium) was used, based on the OWL 2 (Web Ontology Language) Manchester specification, with application examples in [7] and [8].

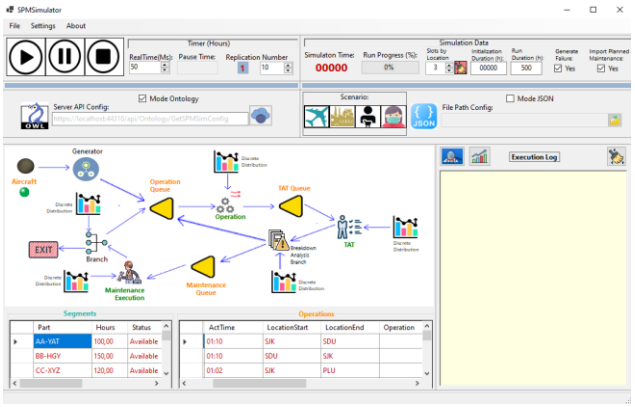


Fig. 2. SPMSimulator interface simulating a line maintenance hangar. Source: the authors.

Through the simulations, it is possible to obtain the mean time between failures (MTBF), the mean time to repair (MTTR), the mean waiting time in the maintenance queue, and the aircraft downtime until the return to operation, using as a basis the information available from the ontologies to feed the simulation model.

It is also possible to update the ontologies according to the results obtained during the simulation; in this work we demonstrate the status of the aircraft displayed on a panel as shown in Fig. 2.

In addition, through the SPARQL queries it was possible to obtain requirements such as the operation data of the air network (see Table II), the history of failure events (see Table III), and the next preventive maintenance actions predicted by the aircraft manufacturers (see Table IV), which serve as input for running the simulation.

Table II. Flight-network data per aircraft.

Activity Time	Location Start	Location End	Date Time Start	Date Time End	Segment
02:45	AJU	GRU	22/05/2024 15:25	22/05/2024 18:10	PP-LAT
01:45	BSB	VCP	22/05/2024 20:40	22/05/2024 22:25	PR-AZL
01:35	VCP	BSB	22/05/2024 13:10	22/05/2024 14:45	PR-AZL
01:10	FLN	CGH	22/05/2024 12:05	22/05/2024 13:15	PP-GGY

Table III. Failure-event history of the assets.

Asset Name	Segment	Event Type	Event Start	Event Description	MTTR
Honeywell GTCP36-100 APU	PP-GGY	Failure Effects	23/05/2024 13:11	Overheating	2 hours
ENGINE CFM56	PP-LAT	Abnormal Situation	23/05/2024 17:47	Engine overload	10 hours

Table IV. Predicted maintenance for the aircraft model.

Asset Name	Unit Type	Execution Time	Segment	Task	MTTR
ENGINE CFM56	Hours	24	PP-LAT	Valve inspections	2 hours
Honeywell APU	Hours	32	PP-GGY	Filter replacement	3 hours

It can be observed that this approach enables the joining of information between the concepts of operations and maintenance requirements in a single query, making it possible to obtain other indicators capable of guiding strategic decision-making, improving resource allocation, planning preventive maintenance, and reducing operational costs.

Another observed gain was that even when changing the data of the aeronautical maintenance hangar ontology to a production line that uses machines in its operation and other types of assets, it was possible to simulate and obtain the same indicators for the new scenario, confirming the proposed interoperability, as demonstrated in Fig. 3.

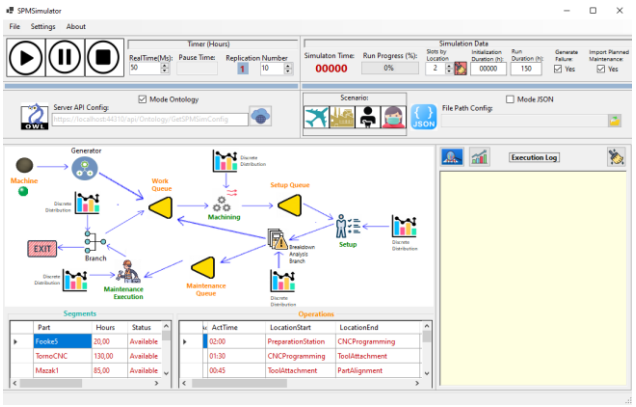


Fig. 3. Simulation using the ontology for a production line. Source: the authors.

For the simulation with the SPMO loaded with data from a production line, the SPARQL queries returned the machine cycle-time information (see Table V), the history of failure events (see Table VI), and the next preventive maintenance actions for the machines (see Table VII), which serve as input for running the simulation.

Table V. Machine cycle-time data.

Activity Time	Location Start	Location End	Date Time Start	Date Time End	Segment
70:00	Preparation Station	CNC Programming	02/05/2024 06:00	02/05/2024 04:00	Fooke5
40:00	Part Alignment	Rough Machining	02/05/2024 06:00	02/05/202 4 22:00	Haas ST

Table VI. Failure-event history of the machines.

Asset Name	Segment	Event Type	Event Start	Event Description	MTTR
Haas ST20Y CNC Lathe	Haas ST	Failure Effects	03/05/2024 10:42	Overheating	10 hours
Fooke CNC Milling Machine	Fooke5	Failure Mode	04/05/2024 15:22	Cooling-system problems	4 hours

Table VII. Preventive maintenance of the machines.

Asset Name	Unit Type	Execution	Segment	Task	MTTR
5-axis Fooke CNC Milling Machine	Hours	32	Haas ST	Lubrication	3 hours

5. Conclusion

With the structuring of the proposed ontologies, it was possible to perform knowledge representation and to carry out inference among elements from different domains. Adding to this the use of SPARQL queries in discrete-event simulations, several benefits arise, such as:

- Interoperability: ontologies provide a standardized and semantic representation of the concepts and relationships in the Maintenance 4.0 domain, facilitating interoperability between systems and applications and allowing different data sources and tools to be integrated more efficiently.
- Data standardization: ontologies provide a common structure to model and organize the relevant simulation data. This helps standardize the data representation, ensuring consistency and reuse across different contexts. With data standardization, it is easier to perform comparative analyses and obtain consistent insights from the simulations.
- Advanced queries: the use of SPARQL queries makes it possible to perform complex and flexible queries on the data modeled by the ontology, allowing the retrieval of specific information, the computation of aggregations, the filtering of data, and inferences about the relationships among the concepts.
- Knowledge discovery: ontologies and SPARQL queries make it possible to explore the knowledge contained in the simulation data. Through inferences and semantic relationships defined in the ontology, it is possible to discover new patterns, identify hidden relationships, and perform more sophisticated analyses, contributing to decision-making in Maintenance 4.0; and
- Flexibility and scalability: the use of ontologies and SPARQL queries offers flexibility and scalability to discrete-event simulations. The ontology can be expanded and updated as new concepts and relationships arise in the context of Maintenance 4.0.

In summary, these advantages contribute to a deeper and more effective analysis of the simulation results, providing valuable insights to improve maintenance operations and decision-making in the context of Maintenance 4.0. The ontology can be used by automated tools to improve advanced services such as web searches and knowledge management, and it can be expanded to other industry contexts.

The proposed ontologies can be used for applications in the 4.0 hangar to support the decision-making of maintenance planners and managers.

As future work, we can mention:

- Including new ontologies relating different contexts, for example, in the field of civil engineering;

- Including new rules to address resource allocation and available manpower by specialty; and
- Improving the simulator to obtain new health indicators for aircraft and assets.
- Implementing real-time data collection through sensors to feed the ontological model with real data.

Conflicts of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Author contributions

M. T. R. da Silva: Conceptualization, Methodology, Software, Writing – original draft; **H. C. Marques**: Conceptualization, Supervision, Writing – review and editing. All authors have read and agreed to the published version of the manuscript.

Data availability

The data that support the findings of this study are not publicly available due to confidentiality agreements with the participating companies, but are available from the corresponding author upon reasonable request.

Funding information

This research received no external funding.

Acknowledgment

The authors would like to thank the commercial aviation and aircraft-manufacturing companies, as well as the maintenance and operations specialists, who contributed information and interviews during the development of this research.

References

[1] H. C. Marques, F. T. M. Abrahão, and G. C. Rocha, Eds., "MRO 4.0: Mapping Challenges Through the ILS Approach," in Applications and Challenges of Maintenance and Safety Engineering in Industry 4.0, Advances in Civil and Industrial Engineering. IGI Global, 2020, pp. 213–234. doi: 10.4018/978-1-7998-3904-0.ch012.

[2] A. Giacotto, H. Costa Marques, E. A. Pereira Barreto, and A. Martinetti, "The Need for Ecosystem 4.0 to Support Maintenance 4.0: An Aviation Assembly Line Case," Applied Sciences, vol. 11, no. 8, p. 3333, Apr. 2021. doi: 10.3390/app11083333.

[3] L. A. Hadidi, U. M. Al-Turki, and A. Rahim, "Integrated models in production planning and scheduling, maintenance and quality: a review," International Journal of Industrial and Systems Engineering, vol. 10, no. 1, p. 21, 2012. doi: 10.1504/IJISE.2012.044042.

[4] M. Yahya, J. G. Breslin, and M. I. Ali, "Semantic Web and Knowledge Graphs for Industry 4.0," Applied Sciences, vol. 11, no. 11, p. 5110, May 2021. doi: 10.3390/app11115110.

[5] H. Hossayni, I. Khan, M. Aazam, A. Taleghani-Isfahani, and N. Crespi, "SemKoRe: Improving Machine Maintenance in Industrial IoT with Semantic Knowledge Graphs," Applied Sciences, vol. 10, no. 18, Art. no. 6325, Jan. 2020. doi: 10.3390/app10186325.

[6] M. A. Musen, "The Protégé project: a look back and a look forward," AI Matters, vol. 1, no. 4, pp. 4–12, Jun. 2015. doi: 10.1145/2757001.2757003.

[7] M. Horridge and P. F. Patel-Schneider, "OWL 2 Web Ontology Language Manchester Syntax (Second Edition)," World Wide Web Consortium (W3C), p. 9, 2012.

[8] M. Unni and K. Baskaran, "Ontology based Semantic Querying of the Web using Protégé," International Journal of Computer Applications, vol. 56, no. 6, pp. 12–16, Oct. 2012. doi: 10.5120/8894-2915.

[9] S. G. García and M. G. García, "Industry 4.0 implications in production and maintenance management: An overview," Procedia Manufacturing, vol. 41, pp. 415–422, 2019. doi: 10.1016/j.promfg.2019.09.027.

[10] H. Marques and A. Giacotto, "Prescriptive Maintenance: Building Alternative Plans for Smart Operations," in Proceedings of the 10th Aerospace Technology Congress, Stockholm, Sweden, Oct. 2019, pp. 231–236. doi: 10.3384/ecp19162027.

[11] K. Matyas, T. Nemeth, K. Kovacs, and R. Glawar, "A procedural approach for realizing prescriptive maintenance planning in manufacturing industries," CIRP Annals, vol. 66, no. 1, pp. 461–464, 2017. doi: 10.1016/j.cirp.2017.04.007.

[12] Q. Z. Yang and Y. Zhang, "Semantic interoperability in building design: Methods and tools," Computer-Aided Design, vol. 38, no. 10, pp. 1099–1112, Oct. 2006. doi: 10.1016/j.cad.2006.06.003.

[13] R. Mizoguchi, "Tutorial on ontological engineering Part 2: Ontology development, tools and languages," New Generation Computing, vol. 22, no. 1, pp. 61–96, Mar. 2004. doi: 10.1007/BF03037281.

[14] M. E. V. Pickler, "Web Semântica: ontologias como ferramentas de representação do conhecimento," Perspectivas em Ciência da Informação, vol. 12, no. 1, pp. 65–83, Apr. 2007. doi: 10.1590/S1413-99362007000100006.

[15] S. Heiler, "Semantic interoperability," ACM Computing Surveys, vol. 27, no. 2, pp. 271–273, Jun. 1995. doi: 10.1145/210376.210392.

[16] OpenO&M, "What is MIMOSA?" Accessed: Jun. 01, 2023. [Online]. Available: https://www.mimosa.org/

[17] B. K. Choi and D. Kang, Modeling and Simulation of Discrete Event Systems. Hoboken, NJ: Wiley, 2013.

[18] L. Chwif and A. C. Medina, Modeling and Simulation of Discrete Events: Theory and Applications (in Portuguese), 3rd ed. São Paulo: Ed. dos Autores, 2010.

[19] J. J. Carroll, I. Dickinson, C. Dollin, D. Reynolds, A. Seaborne, and K. Wilkinson, "Jena: implementing the semantic web recommendations," in Proceedings of the 13th International World Wide Web Conference on Alternate Track Papers & Posters, New York, NY, USA: ACM Press, 2004, p. 74. doi: 10.1145/1013367.1013381.

[20] S. Harris and A. Seaborne, "SPARQL 1.1 Query Language," World Wide Web Consortium (W3C), Mar. 2013. [Online]. Available: https://www.w3.org/TR/2013/REC-sparql11-query-20130321/

[21] Apache Software Foundation, "Apache Jena," 2021. [Online]. Available: https://jena.apache.org