Development of an Application Ontology for Knowledge Management to Support Aircraft Assembly System Design

Xiaochen Zheng[1], Jinzhi Lu[1], Rebeca Arista[2], Xiaodu Hu[3,4], Joachim Lentes[3], Fernando Ubis[5], Jyri Sorvari[5] and Dimitris Kiritsis [1]

[1] *École Polytechnique Fédérale de Lausanne (EPFL), Lausanne 1015, Switzerland*
[2] *Airbus SAS, Blagnac 31700, France.*
[3] *Fraunhofer IAO, Nobelstrasse 12, 70569, Germany*
[4] *University Stuttgart IAT, Nobelstrasse 12, 70569, Germany*
[5] *Visual Components, Espoo 02600, Finland*

Abstract

During the early phase of an aircraft program, industrial architects need to evaluate different industrial scenarios and perform trade-offs to optimize the future industrial architecture according to different key performance indicators. Expert knowledge accumulated during previous programs provides foundation for the new one. It is a challenging task to capture and reuse expert knowledge in a consistent way. This paper presents a case study about the development of a formal application ontology for aircraft assembly processes. It aims to facilitate expert knowledge capturing from existing programs and reusing it to support new aircraft assembly system design. This application ontology inherits the structure and classes from the IOF-Core ontology as the basis, which adopts BFO as the top-level ontology. Historical assembly process specifications and domain experts' feedbacks are used as knowledge sources of the ontology. Relevant elements of the assembly process including all the operations, materials and manufacturing resources are extracted and integrated into the ontology as individuals. Based on the analysis of these individuals, common knowledge which can be reused in similar processes can be generalized as interrelated classes of the ontology. The detailed development approach of the application ontology is introduced using an industrial pilot. The developed ontology is integrated as the core functional block of a trade space framework. It can help track stakeholders' requirements and support co-simulations of the new assembly process.

Keywords

Ontology, IOF, BFO, aircraft assembly, knowledge management, systems engineering, ontology-based engineering.

1. Introduction

The assembly lines in the aerospace industry are highly complex systems characterized by a low-scale manufacturing rate, producing mid to high level customized products [1]. They are different from other mass production industries that produce multiple standard products in the same assembly line, with a medium or high-scale manufacturing rate. During the early phase of an aircraft program, industrial architects need to evaluate different industrial scenarios and to perform trade-offs to optimize the future industrial architecture according to different performance parameters like labor cost, industrial assets cost, lead time and different kind of defect risks within the extended enterprise. The

FOMI 2021: 11th International Workshop on Formal Ontologies meet Industry, held at JOWO 2021: Episode VII The Bolzano Summer of Knowledge, September 11–18, 2021, Bolzano, Italy
EMAIL: xiaochen.zheng@epfl.ch (X. Zheng); jinzhi.lu@epfl.ch (J. Lu); rebeca.arista@airbus.com (R. Arista); xiaodu.hu@iao.fraunhofer.de (X. Hu); joachim.lentes@iao.fraunhofer.de (J. Lentes); fernando.ubis@visualcomponents.com (F. Ubis); jyri.sorvari@visualcomponents.com (J. Sorvari); dimitris.kiritsis@epfl.ch (D. Kiritsis)
ORCID: 0000-0003-1506-3314 (X. Zheng); 0000-0001-5044-2921 (J. Lu)

aircraft assembly lines are usually not efficiently flexible to new manufacturing scenarios or new product developments. Therefore, one of the key tasks for a new conceptual design is to correctly model and specify the assembly system baseline definition and flexibility [1]. Such baselines, which are based on existing assembly system setups under production phase or other predecessors, provide a solid starting point for the new system design. It leads to the industrial requirement of capturing and reusing existing knowledge.

Knowledge about existing systems is intangible and varies among different experts. It is crucial to capture and instantiate expert knowledge in a persisting way so that it can be reused in a tool-agnostic way for new programs. Ontology is a powerful tool for knowledge management which enables capturing information about the world that is compatible with the perspective of human common sense. It can be used as a reference schema providing a unified and coherent view over existing systems. Ontology has been widely applied in many fields for knowledge management, among which manufacturing is one of the main application areas. Aiming at creating a common semantic net for manufacturing domain, Lemaignan et al. [2] developed the MASON (MAnufacturing's Semantics ONtology) as a preliminary upper ontology for manufacturing. Negri et al. [3] developed the MSO (Manufacturing System Ontology) which models the discrete manufacturing, process production and the logistics domains. Sange et al. [4] developed the Z-BRE4K ontology as the core of a semantic-driven approach for realizing Zero Defect Manufacturing (ZDM). Foehr et al. [5] the GRACE Ontology focusing on quality optimization of discrete manufacturing processes. They use the MPFQ-model (Material, Production Processes, Product Functions/Features, Product Quality) to organize the main factors during production that can impact product quality. Many more ontologies can be found which indicates the tend of using ontology to facilitate knowledge management in manufacturing domain including the aerospace production sector. A previous study [6] has explored the feasibility of using industrial ontology to support aerospace assembly line design process.

Despite the popularity of ontology applications in manufacturing, there are relatively less application cases reported compared with other mass-production sectors. This study presents an application ontology for knowledge capturing and reusing to support assembly system design during the early phase of a new aircraft program. We first introduce the methodology which guides the development of the ontology including the adopted top-level ontology BFO and IOF-Core ontology. Then we demonstrate the structure and main elements of the application ontology; and in the end, we discuss how this ontology can be used to support requirement tracking and trade-off simulations.

2. Methodology

Ontology development has become an engineering discipline containing a set of activities that concern the ontology development process and lifecycle, the methods and methodologies for building ontologies, and the tool suites and languages that support them [7]. The results of ontology engineering provide formal domain knowledge representation to be reused efficiently and prevent waste caused by non-shared knowledge, as well as improve interoperability and standardization. Ontology can play different roles simultaneously for knowledge management [8] such as trusted source of knowledge, knowledge base, cross-domain bridge, mediator for interoperability, contextual search and linked data enabler, among others.

After decades of development, numerous ontologies have been created based on various application scenarios using different languages and tools. This diversity brings obstacles for the integration of different ontologies in a unified framework to assure their interoperability and reusability. A possible solution for this problem is to make use of a hierarchical methodology to unify the application ontologies under a common top-level ontology which contains a set of general vocabularies commonly used across all domains. These vocabularies are properly structured and formally defined. Such top-level ontologies provide a common foundation for developing lower-level ontologies such as domain-specific ontologies and more detailed application ontologies. The adoption of the top-level ontology assures semantic interoperability among these lower-level ontologies. Currently, many top-level ontologies have been developed and widely applied by different communities such as Basic Formal Ontology (BFO) [9] and Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) [10].

Some recent efforts have been spent on unifying and standardizing existing domain ontologies based on certain top-level ontologies. For example, the Industrial Ontologies Foundry (IOF) [11] which is an ongoing initiative aims to co-create a set of open ontologies to support the manufacturing for industrial needs and to promote data interoperability. IOF provides a multi-layer architecture to guide ontology development, consisting of four layers: top-level foundation ontology, domain-level (domain independent and domain specific) reference ontologies, subdomain ontologies and application ontologies. It uses BFO as a foundation involving experts from different industrial domains work jointly to create open and principles-based ontologies.

In this study, we follow the IOF principles and use the IOF-Core ontology as the basis to develop our application ontology for the assembly process. IOF provides the IOF-Core ontology [12] which contains top terms that can be used as starting point for creating application ontologies. The definitions and properties of the classes are available in the core ontology which is open access. IOF-Core ontology refers to BFO as top-level ontology. The structure of BFO is based on a division of entities into two disjoint categories of continuant and occurrent, the former comprehending objects and spatial regions, the latter comprehending processes conceived as extended through (or as spanning) time [9]. Like BFO, the IOF-Core ontology also categorizes entities into *continuant,* which is "*persists, endures, or continues to exist through time while maintaining its identity*", and *occurrent*, which "*unfolds itself in time or it is the start or end of such an entity or it is a temporal or spatiotemporal region"*. Some of the main classes of the IOF-Core ontology are shown in Figure 1.

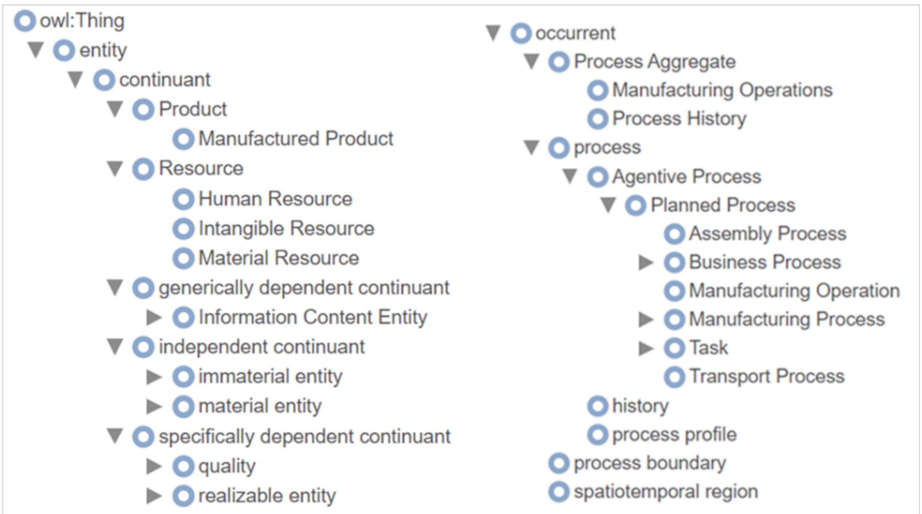

Figure 1 Main classes of the IOF-Core ontology

3. Development of Application Ontology

Application-level ontology is the lowest ontological level which aims to represent specific application cases with highly specialized classes and individuals, such as a device, a part, a component, an equipment etc. of the assembly line. The application ontology uses or refers top-level or domain ontologies to construct ontological classes and relationships between classes. In this study we use the IOF-Core ontology as the foundation for the application ontology and add customized classes and individuals according the requirements of the aircraft assembly process which is introduced in the following section.

The main benefits of using the IOF-Core ontology is to support knowledge capturing in a formal and persistent way thus to assure both intra- and inter-organization interoperability. More specifically, this ontology is used to align the existing knowledge from different stakeholders, including the external technology providers (e.g. 2D and 3D simulation), and internal experts (e.g. industrial system engineer, industrial architect). Moreover, in this case three knowledge sources are defined including domain experts, historical documents and public resources like research papers, technical reports etc. IOF-Core ontology provides a criterion to align and integrate all the relevant knowledge.

3.1. Application Scenario

The application ontology is developed as a key functional block of a trade space framework for aircraft industrial system design. It focuses on the research and development phase of the assembly line for a new model of aircraft. During this early phase, industrial architects need to evaluate different industrial scenarios and to perform trade-off among different performance parameters for the future industrial architecture. More specifically, the application scenario focuses on the fuselage orbital junction process to be designed for a given assembly station of a Final Assembly Line (FAL) for the new aircraft model. There are two options to execute the orbital junction process, i.e. a manual process and an automated process using a flex-track robotic mechanism.

The trade-off is expected to be performed between the manual process and the automated flex-track process. The main differences between them are the external and internal drilling operations. For the manual process, both drilling operations are performed by operators; for the automated option, the external drilling operations are performed by the Flex Track robot while the internal drilling operations are performed by operators.

An overall functional architecture of the trade space framework is defined using a Model Based Systems Engineering (MBSE) approach. As shown in Figure 2, it contains several functional blocks including Requirement Management block, Architecture Definition block, Visualization block, System Integration block and Verification block. The ontology is the core of the System Integration block which integrates all relevant data and information from other blocks.

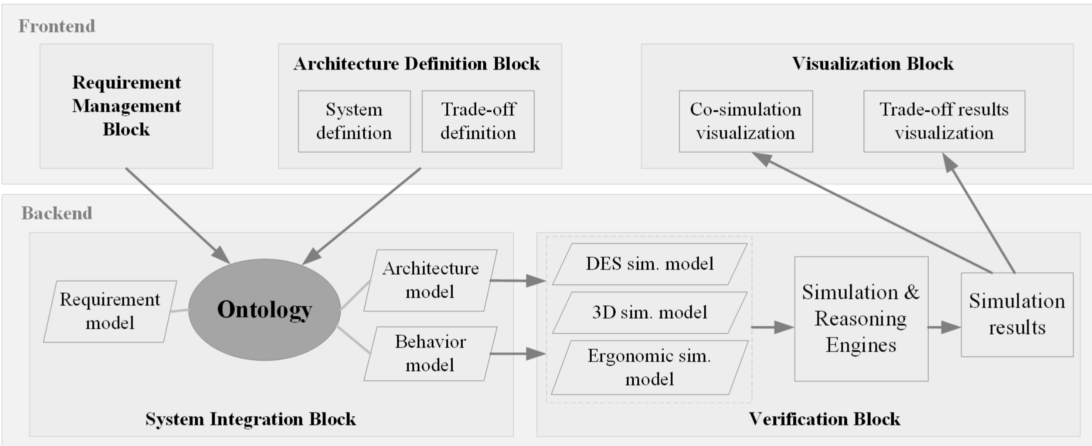

Figure 2 Overview of the functional architecture of trade space framework for aircraft industrial system design

3.2. User Stories and Vocabulary

The main stakeholders for ontology development during the system design phase are the Industrial System Architects and Industrial System Engineers. Prior to ontology development, a survey has been conducted to gather user stories and stakeholders' requirements. The most relevant user stories to ontology development is about knowledge capturing and reusing, which align well with the main functions of ontology in the overall functional architecture of the trade space framework as shown in Figure 2. The corresponding user story is as follows:

- *As an Industrial System Engineer, I want to have a methodology to capture experts' knowledge in a persisting way, in order to have a tool-agnostic Orbital Joint Process definition that allows a ZDM design.*

The keywords of the above user stories for ontology development are *"capture experts' knowledge"* and *"instantiate the knowledge captured"*. *"Orbital Joint Process"* is the name of the application scenario and *"ZDM design"* is the final target of the system which is zero-defect manufacturing.

There are two main sources of experts' knowledge, the knowledge stored in tangible documents and the knowledge stored in experts' mind. The former one can be gathered by analyzing historical data and information of existing assembly systems, and the later needs to be collected through modeling methodologies. In this study, we currently focus on the historical knowledge and keep the interactions with experts as future actions. The process specifications of existing orbital joining processes are one of the main inputs for vocabulary identification. An example of such process specifications is shown in Figure 3.

	Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names
1		Start	0 mins	Mor	Mon		
2		: Set up working envirionment	10 mins	Mor	Mon		C35 Upper_1
3		: Set up of wedge left side	10 mins	Mor	Mon	2	C35 Upper_1
4		: Attaching L18	10 mins	Mor	Mon	3	C35 Upper_1
5		: Set in position Rails and LFT	15 mins	Mor	Mon	4	C35 Upper_1
6		: PROBE_1	5 mins	Mor	Mon	5	C35 Upper_1
7		: Drilling orbital 4,8(1)	30 mins	Mor	Mon	6	C35 Upper_1
8		: PROBE_2	5 mins	Mor	Mon	7	C35 Upper_1
9		: Drilling orbital 4,8(2)	30 mins	Mor	Mon	8	C35 Upper_1
10		: Drilling template install 18G to 22G, bore	30 mins	Mor	Mon	9	C35 Upper_1

Figure 3 Part of an orbital joining process specification

3.3. Ontology Individuals and Classes

As shown in Figure 3, the orbital joining process consists of a series of tasks. Each task is composed of certain operations and requires different materials and resources. It also includes the duration of each task and its predecessors. Each of these tasks is analyzed to extract vocabularies and their relationships for the ontology. A corresponding individual is created in the application ontology for each task containing its relevant information. The *"task name"* indicates the operation type and involved materials; the *"duration"* is mapped to the *"op_duration"* property of the individual; the *"predecessors"* specifies the sequence of the individual and the *"resource names"* indicates the necessary resources to conduct this task. As shown in in Figure 4, the task *"S40_012_Set in position temporary fastener 3,2 into buttstrap (1)"* contains three main elements: *"Set in position"* indicates this task involves a *positioning operation*, which is an operation class of the ontology; *"temporary fastener 3.2"* and *"buttstrap (1)"* indicate the two materials needed to finish this task. These elements are inserted into the ontology as new individuals if they are not existing.

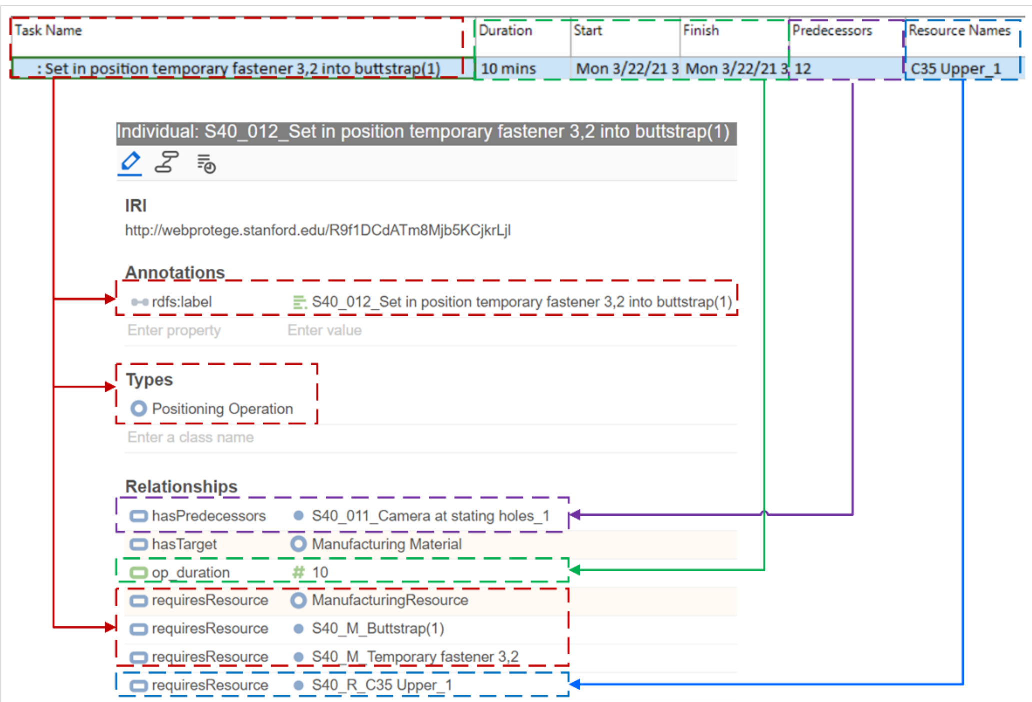

Figure 4 Exact information from process specifications to create ontology individuals

When creating new individuals, some individuals can be reused and some of them belong to the same type. For example, in the tasks, different *"Butterstrap"* individuals are mentioned like

“Butterstrap4.8”, *“Butterstrap (1)”* and *“Butterstrap (2)”*. Obviously, they can all be categorized as *“Butterstrap”*, thus a corresponding class is created in the ontology as shown in Figure 5. In this way, all the individuals can be assigned to a corresponding class with predefined properties. When a new individual is created, it can directly inherit the predefined properties and relationships, as shown in Figure 5.

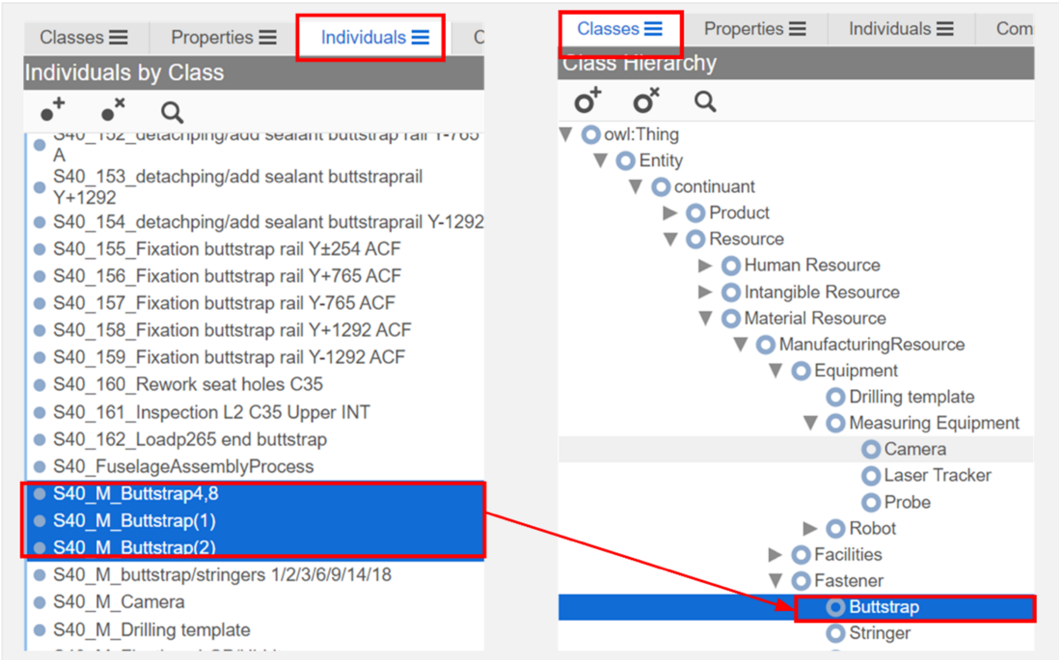

Figure 5 Generalizing individuals to create new classes

To formalize the captured knowledge from the individuals, it is necessary to analyze the relationships between frequently used classes. The RMPFQ (Resource, Material, Processes, Functions/Features, Quality) model [13,14] is used to organize the relevant classes and specify their relationships. This model categorizes the factors that can affect product quality during manufacturing and specifies their relationships. As shown in Figure 6, there are several types of interrelations among the elements of the RMPFQ-model. For an assembly process, multiple *Materials* are assembled through a planned machining *Process*, which requires certain *Resources*. This composes the RPM interaction (marked with orange lines) which is the focus of this study. Other interactions involving Functions/Features and Quality are introduced in existing studies [13,14].

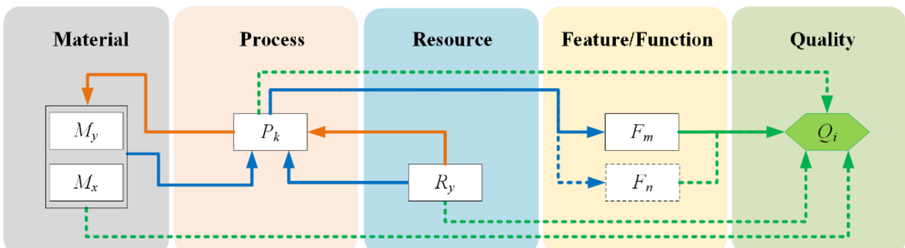

Figure 6 RMPFQ-model elements and their interrelations

The case of this study covers mainly the resource, material and process factors, whereas the function factor is covered at a higher system level which is out of the scope of this paper. Relevant classes are added to the IOF-Core ontology. Figure 7 shows some of the classes added to the IOF-Core ontology corresponding to the resource, material and process of application case.

The *“Orbital Joining Process” is_a “Assembly Process (P)”*. The *“Orbital Joining Process” joins “Front Fuselage”* and *“Rear Fuselage”* which are *“Manufacturing Materials (M)”*. To execute the *“Orbital Joining Process”*, it *requires* a set of *“Manufacturing Resources (R)”* such as *“Equipment”* and *“Facilities”*.

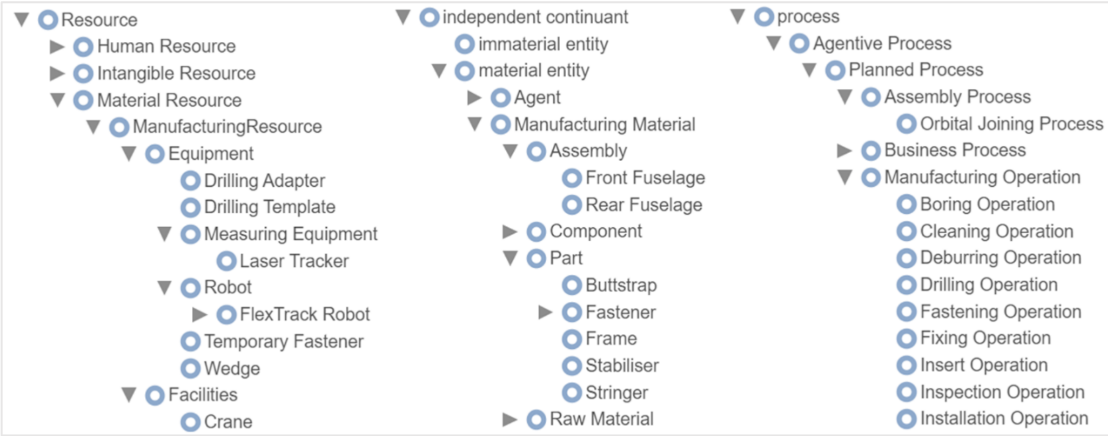

Figure 7 Part of the classes added to IOF-Core ontology

An individual of the ontology is an instance of one the classes and inherits its properties and relationships with other classes. Therefore, all the individuals related to the orbital joining process can be interconnected. As shown in Figure 8, the individual *"S40_OrbitalJointProcess"* represents the existing orbital joining process which provides the historical knowledge. This process consists of a series of operations (162 operations in this case). The *"hasOperation"* property assigns all the individuals corresponding to these operations, such as *"S40_013_Drilling buttstrap 4.8"*, to the *"S40_OrbitalJointProcess"*. Each of these operations requires relevant materials and resources to execute. For example, in Figure 8, the *"S40_013_Drilling buttstrap 4.8"* operation *requiresMaterial "S40_M_Buttstrap4.8"*, and *requiresResource "S40_R_C35 Upper_1"*. For each manufacturing resource and material, it has its own properties representing the constrains and characteristics which can be used to support trade-offs in later steps. For instance, the "*S40_R_C35 Upper_1"* in Figure 8 has properties like *"hasBaseCalender"*, *"hasCostPerUse"* and *"hasOvt.Rate"* etc., which represents the availability, cost and efficiency of this resource.

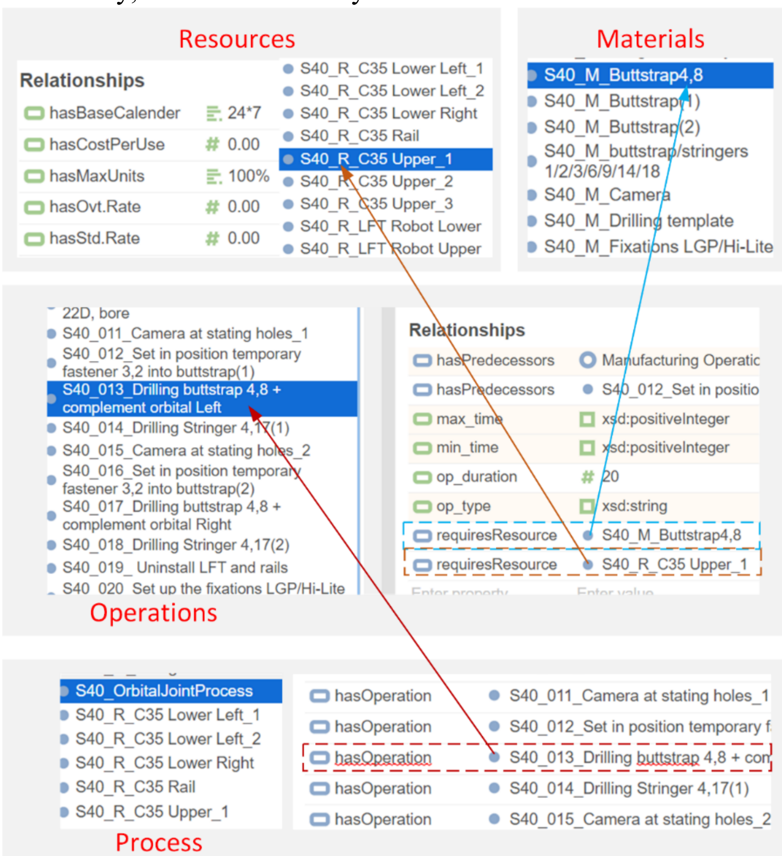

Figure 8 Interrelationships between ontology individuals

4. Application and Results

As introduced previously in the application scenarios, the application ontology is the core of the system integration functional block of the trade space framework. It is used to integrate information and knowledge about the requirements, architecture and behavioral models, and process specifications. On the other hand, it provides necessary input for the verification block which includes simulation and reasoning. To facilitate the integration, in addition to the classes and individuals corresponding the orbital joining process, extra classes and individuals representing system requirements, architecture and behavior models are added to the application ontology by requirement management and architecture definition block correspondingly.

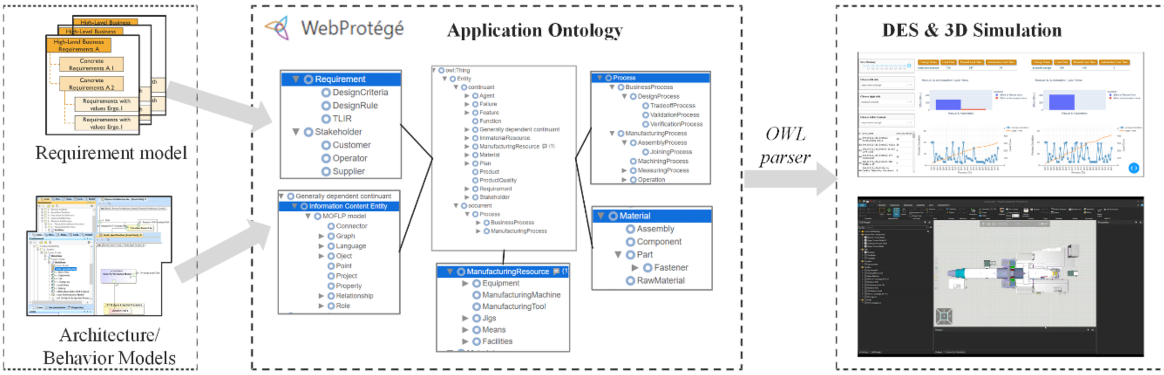

Figure 9 The role of ontology in the trade space framework and its connections with other function blocks

As shown in Figure 9, the ontology serves as the information and knowledge integration hub providing input for the discrete event simulation (DES) and 3D simulation in the verification block. The ontology is exported as OWL file and a customized OWL parser is developed to parse the ontology and extract necessary information for the simulations. The parsing method and simulation process will be introduced in separate studies which is beyond the scope of this paper. The simulation results are further processed and visualized by the visualization function block to support decision making.

5. Discussion

The work presented in this paper is part of an on-going project and the ontology and relevant function blocks are under frequent updating. It aims to demonstrate the basic workflow of applying ontology to support knowledge capturing and reusing thus to enable a trade space framework for aircraft industrial system design. There are several pending issues which need to be addressed before fully realize the expected knowledge management target.

The knowledge captured from the existing assembly processes is only a part of the knowledge sources which is based on the tangible documents like historical process specifications. The individuals and classes of the current version ontology need to be optimized with more clear structure and definitions. There will be two more knowledge sources to be included in the future. First, the knowledge of domains experts which is intangible and will need deep discussions and interviews with domains experts to complete and formalize the captured knowledge. Some tools and software will be used to transcribe the experts' knowledge. Another source is the public research papers, technical reports and white papers etc. Knowledge gathered from such public sources will be used as references and complements for the previously developed ontology.

As mentioned in the application and results section, the ontology includes not only the process knowledge, but also the requirement, system architecture and behavior models. In the current version of the ontology, corresponding classes and individuals have been added, but they are not yet fully integrated. The relationships between these classes and individuals, and the ones corresponding to the process operations have not fully defined. This hinders the reasoning function of the verification block. Efforts from requirement management experts and system architects together with ontology experts are being spent to complete the definition of these relationships.

The aircraft industrial systems are highly complex systems. The orbital joining process introduced in this study is only one process of the assembly line. The aim of this study is to verify the feasibility of ontology applications. More efforts are required to expand the ontology to include more processes of the industrial system to create a complete application ontology.

6. Acknowledgements

The work presented in this paper is funded by the EU H2020 project QU4LITY (825030) - Digital Reality in Zero Defect Manufacturing, and the EU H2020 project OntoCommons (958371) - Ontology-driven data documentation for Industry Commons.

7. References

[1] R. Arista, F. Mas, M. Oliva, D. Morales-Palma.Applied ontologies for assembly system design and management within the aerospace industry, in: CEUR Workshop Proc., 2019.

[2] S. Lemaignan, A. Siadat.MASON: A proposal for an ontology of manufacturing domain, in: IEEE Work. Distrib. Intell. Syst. Collect. Intell. Its Appl., 2006: pp. 195–200.

[3] E. Negri, L. Fumagalli, M. Macchi, M. Garetti.Ontology for service-based control of production systems, in: IFIP Adv. Inf. Commun. Technol., Springer New York LLC, 2015: pp. 484–492.

[4] S. Cho, G. May, D. Kiritsis.A semantic-driven approach for industry 4.0, in: Proc. - 15th Annu. Int. Conf. Distrib. Comput. Sens. Syst. DCOSS 2019, Institute of Electrical and Electronics Engineers Inc., 2019: pp. 347–354.

[5] M. Foehr, T. Jäger, C. Turrin, P. Petrali.Implementation of a methodology for consideration of product quality within discrete manufacturing, in: IFAC Proc. Vol. 46(9), 2013: pp. 863–868.

[6] R. Arista, F. Mas, C. Vallellano.Initial Approach to an Industrial Resources Ontology in Aerospace Assembly Lines, in: IFIP Adv. Inf. Commun. Technol., Springer Science and Business Media Deutschland GmbH, 2020: pp. 285–294.

[7] A. Gómez-Pérez, M. Carmen Suárez-Figueroa.NeOn Methodology for Building Ontology Networks: a Scenario-based Methodology, 2009.

[8] S. El Kadiri, D. Kiritsis.Ontologies in the context of product lifecycle management: state of the art literature review. Int. J. Prod. Res.; 2015; 53: 5657–5668.

[9] R. Arp, B. Smith.Function, Role, and Disposition in Basic Formal Ontology. Nat. Preced.; 2008; 1–1.

[10] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, A. Oltramari, I. Horrocks.IST Project 2001-33052 WonderWeb: Ontology Infrastructure for the Semantic Web, 2003.

[11] Industry Ontology Foundry (IOF). Technical Principles – IOF Website. 2021; URL: https://www.industrialontologies.org/technical-principles/.

[12] Industry Ontology Foundry (IOF). GitHub - NCOR-US/IOF-BFO at IOF-Core-2020. 2021; URL: https://github.com/NCOR-US/IOF-BFO/tree/IOF-Core-2020

[13] X. Zheng, F. Psarommatis, P. Petrali, C. Turrin, J. Lu, D. Kiritsis.A quality-oriented digital twin modelling method for manufacturing processes based on a multi-agent architecture, in: Procedia Manuf., Elsevier B.V., 2020: pp. 309–315.

[14] X. Zheng, J. Lu, P. Petrali, C. Turrin, D. Kiritsis.A Semantic-Driven Digital Twin Model for Machining Processes Towards Zero Defect Manufacturing. 2021; URL: https://www.researchgate.net/publication/352822634_A_Semantic-Driven_Digital_Twin_Model_for_Machining_Processes_Towards_Zero_Defect_Manufacturing.