



**HAL**  
open science

# Scalable Ontology-Based V&V Process for Heterogeneous Systems and Applications

Romain Delabeye, Olivia Penas, Régis Plateaux

► **To cite this version:**

Romain Delabeye, Olivia Penas, Régis Plateaux. Scalable Ontology-Based V&V Process for Heterogeneous Systems and Applications. MODELS '22: Proceedings of the 25th International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings, Oct 2022, Montréal, Canada. 10.1145/3550356.3561577 . hal-03912273

**HAL Id: hal-03912273**

**<https://hal.science/hal-03912273v1>**

Submitted on 19 Jan 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Scalable Ontology-Based V&V Process for Heterogeneous Systems and Applications

Romain Delabeye  
Quartz Laboratory, EA7393  
ISAE-Supméca  
Saint-Ouen, France  
romain.delabeye@isae-supmecca.fr

Olivia Penas<sup>†</sup>  
Quartz Laboratory, EA7393  
ISAE-Supméca  
Saint-Ouen, France  
olivia.penas@isae-supmecca.fr

Régis Plateaux  
Quartz Laboratory, EA7393  
ISAE-Supméca  
Saint-Ouen, France  
regis.plateaux@isae-supmecca.fr

## ABSTRACT

This work focuses on ongoing research within the EU-funded EnerMan project aiming at improving the energy efficiency of manufacturing systems. Industrial use cases are generally too constrained to easily proceed to the verification and validation (V&V) of the scientific approaches tackling their challenges. In this context, we propose an ontology-based framework with a methodology assessing the scalability of heterogeneous systems, environments, and missions in a V&V context. Indeed, projecting these industrial and laboratory applications onto a meaningful ontology allows them to be flattened out to the same scale from a semantic point of view. Reasoning is used to evaluate the extent to which a given scientific approach can be verified on a laboratory use case different from the industrial scenario on which it has to be validated. The framework has been implemented using Protégé and Owlready2, and applied to a scientific approach focused on a blind source separation technique used to identify system operating modes in a black box manner, tested on a coffee machine and two industrial case studies (a vehicle testbed's heating ventilation and air conditioning system, and a chocolate production line).

## CCS CONCEPTS

• Knowledge representation and reasoning • Semantic web description languages • Verification and validation

## KEYWORDS

Verification & Validation, Ontology, Scalability, Heterogeneous Systems, Requirements Analysis

## 1 Introduction

The main objective of the EnerMan Project (2021-2024) is to

© 2022 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by an employee, contractor or affiliate of a national government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only.

MODELS '22 Companion, October 23–28, 2022, Montreal, QC, Canada

© 2022 Association for Computing Machinery.

ACM ISBN 978-1-4503-9467-3/22/10...\$15.00

<https://doi.org/10.1145/3550356.3561577>

define an energy sustainability management framework in which plant data is collected and processed holistically to optimize energy sustainability. In this collaborative project, each model or approach developed by technical partners should be validated on industrial use cases. Given the geographical distance and the limited availability of the consortium's industrial systems (production systems under operation), it is usual to test scientific approaches in a laboratory environment first, with a system that is available, open, upgradeable, and easily instrumented as necessary. In this context, two key questions arise: (i) how to select a laboratory system that is representative of the industrial end-user's system with respect to a given scientific Approach To be Validated (ATV) (ii) how to verify and validate scientific approaches on heterogeneous systems (of different sizes, complexity levels, physical characteristics and operational constraints)? The case of a laboratory versus an industrial system is investigated in this paper; the question extends to heterogeneous industrial use cases though. In other words, given the scope of the project under consideration, an ambitious challenge could be: how to validate the energy sustainability of a manufacturing system with a coffee machine?

To answer the first question, a unified framework is required. We therefore designed the WONKA (Verification & Validation with an ONtology and Knowledge-based Approach) framework that hosts a generic semantic representation for sharing structured knowledge from different perspectives and contexts. This framework must also support the analysis of semantic similarity and scalability between two heterogeneous systems. To tackle the second question, verification and validation (V&V) processes need to be taken into account, especially in a model-based system engineering (MBSE) context.

Focusing on our case study, the developments carried out in the EnerMan project have to be applicable to a wide spectrum of manufacturing systems (automotive, food, semi-conductor, metal industries, medical devices, etc.). As a result, the corresponding scientific approaches proposed by the project partners must be generically applicable to a variety of manufacturing systems, with limited information about their architecture or behaviors. In this context, it is interesting to work on black box solutions, with only sensor data as input, which can automatically generate an identity card of the system, containing its behavior (as a model) and necessary information to estimate Energy Performance Indicators

(EnPI). Therefore, the ATV for the proposed case study consists of a blind source separation method (BSS) that identifies and differentiates the operating modes of a black box system and decomposes each mode as an activation mapping of the actuators [12]. This will enable to estimate and predict actuator-specific EnPI, yet we limit the scope of this case study to the preliminary pre-requisite decomposition problem. This technique has to be verified on a coffee machine (lab system) and then validated on two industrial systems, namely a heating, ventilation, and air conditioning (HVAC) system for a vehicle testbed, and a chocolate manufacturing system.

The contribution of this paper consists of a methodology and software for the verification and validation of a scientific approach from a laboratory system to an industrial one, based on an ontological model.

Section 2 will present state-of-the-art V&V processes, focusing on the context of scalability of heterogeneous systems, then on similarity-based approaches. Existing relevant ontologies capable of supporting the systems engineering activities addressed will be mapped. Section 3 will present the WONKA framework developed. Section 4 will detail the verification process on a laboratory system, including its selection for the next ATV validation step. Section 5 will focus on WONKA's validation process on initially identified industrial systems. Methods detailed in sections 4 and 5 are applied on the case study. Conclusions and future work are presented in Section 6.

## 2 Related Work

### 2.1 Scalable Verification and Validation

In systems engineering, V&V processes are used to determine whether the development products of a given activity conform to the requirements of that activity and whether the product satisfies its intended use and user needs [22]. Systems scalability has mostly been studied as a change in size [45] and design maturity [3], or in computer engineering applied to information systems and software focusing on their complexity degree [21]. Scalability or extensibility refers more generally to the ability of a product or system to maintain its functional properties despite a change in size or certain parameters. Following this paradigm, some authors have been interested in the verification of a system when it changes its context of use [36,38], or during software design transformations [4]. Still, scalable V&V has received little attention in the literature: some works relate to computer systems, using formal methods (e.g., model checking) [1,4]. These rarely address physical systems though, mainly focusing on simulation [6,17].

Some authors have worked on the scalability of heterogeneous hardware systems by focusing on their sensing and actuation aspects [32,38]. Still, to our knowledge, the scalability of the V&V of the same scientific approach on heterogeneous systems (with a different structure, various behaviors, and diverse missions) has not been tackled, except when considering their commonalities from a specific point of view.

### 2.2 Similarity-Based Approaches

To exploit the similarity between two physical entities, different approaches have been developed, depending on whether both address the same functionality, share the same behavioral model, or respect the same interaction rules.

Robles et al. [38] define that two devices are close semantically, or have a short semantic distance between them, when the devices have similar functionalities.

A metamodel can be used as the backbone of the methods and process tools to be implemented to fulfill a mission or solve an engineering issue [29]. A metamodel is then considered as a common and sharable reference model that can be used on many systems in many individual situations or configurations, provided that they similarly respect the behavior described by the metamodel [5].

An ontology provides a unifying framework for a shared understanding between various viewpoints whatever the stakeholders' needs and their context. As a result, ontology can have the necessary depth to represent similar objects. According to the classification of Uschold and Gruninger [43], an ontology presents four levels of formalization, of which only the formal level can support deduction mechanisms, in particular for problem solving in the systems engineering domain. Thanks to its generic representation (abstraction level), ontology is useful for the requirements definition, the automation of consistency checking and the reusability of similar concepts in multiple contexts. Some authors have used ontology to ensure verification and validation of physical systems, more specifically for security requirements related to their risk level [41] or their specification process [42].

To answer the two research questions initially posed, as ontologies offer a unified framework that hosts a generic semantic representation that allows sharing structured knowledge from different points of view and contexts, they appear as an appropriate solution to support the analysis of the semantic similarity and the scalability between two heterogeneous systems.

### 2.3 Related Existing Ontologies

Based on the scope of the case study, we have investigated ontologies in which manufacturing systems and their dynamic behavior can be accurately expressed, while supporting V&V processes.

Regarding the energy sustainability, multiple ontologies focus on energy evaluation and management in manufacturing [10,44], product lifecycle management [7] or manufacturing factory sustainability [14,15].

Ontologies allowing for representing dynamic systems include control [32], condition monitoring [2,8] and prognostic [34] aspects, and aim to describe and use the interactions between related sensors [9] and actuators.

The ONTO-PDM [35] ontology uses a product-centric interoperability architecture including business, engineering, and manufacturing aspects. Numerous manufacturing-related ontologies, such as MASON [27] or ExtruONT [37], rather focus on the latter, thus allowing for a detailed expression of production operations, systems, resources and plant layout. These ontologies

put the emphasis on production management, and not on the physical behavior of a plant's constitutive elements. In fact, many of these manufacturing ontologies are aligned with a sensor-centric one, namely the Sensor, Observation, Sample and Actuator (SOSA) ontology [24] or one of its extensions such as SSN and DUL [20]. Together with the SAREF module SAREF4INMA [39], these ontologies accurately describe the physical interactions between manufacturing entities.

Kesadri et al. have proposed a Verification and Validation Ontology (VVO), based on the hierarchy of viewpoints and formalisms concepts, and especially V&V techniques, in order to identify the V&V tools that comply with a V&V technique, based on a given formalism [25].

The suitable ontology required for our scope will be determined depending on the objective and context determined by the use cases.

### 3 The WONKA Framework

The WONKA framework consists of a generic semantic representation designed to formally represent heterogeneous systems and applications, from different perspectives and contexts. This framework was tailored to the V&V processes presented in Figure 1, stemming from the following problem analysis. These processes will be applied to the WONKA case studies.

#### 3.1 Problem Analysis

V&V processes are part of Systems Engineering processes [22,23] and are based on requirements. As a result, it will be important to consider them in our approach. In parallel, V&V processes can be performed either by physical (test bench) or virtual experiments (simulation, analytical resolution). Some verifications can be automated by formal techniques applied to static requirements (model checking, etc.) or with simulation for dynamic requirements [13]. In most cases, the experiments are carried out on a laboratory system, a simulator or prototype, as the real system of interest is more constrained (availability, configurability, upgradeability, etc.). Still, when developing a scientific approach to meet an industrial challenge, the choice of a representative laboratory system onto which the approach will be verified is a crucial issue for researchers. This aspect raises the question of the adequacy of an application case with respect to the intended approach to be validated. Then, validation must eventually be performed on the industrial system in its environment. This boils down to evaluating whether an approach – that has been verified on a laboratory system in a specific environment – could be successfully scaled up on the industrial system and its more constrained environment.

#### 3.2 Approach Overview

The proposed WONKA framework addresses two main objectives: (i) to support the choice of a laboratory system that can verify a given ATV while being representative enough of the

industrial end-user system, (ii) on which the ATV has to be then validated within its application environment.

WONKA includes a methodology based on four main steps: (i) MBSE analysis, (ii) ontology design, (iii) requirement-based query definition, and finally (iv) static and dynamic V&V processes.

First, the industrial challenge (i.e. the issue to be tackled by the industrial system in its environment) has to be analyzed in relation to an ATV, based on systems engineering principles. This requirements elicitation activity is supported by MBSE methods [30,40]. It aims to define the requirements that will have to be verified or validated during the static and dynamic V&V processes.

For the second step, requirements will help the design or selection of an ontology able to capture all relevant aspects of the use cases addressed, including the industrial system(s) (IndSys) and the laboratory system(s) (LabSys).

Third, these requirements – built from the ATV, the industrial challenge, the IndSys and LabSys constraints – have to be transformed into queries based on the ontological concepts and properties, in order to be formally evaluated on the resulting knowledge graph (once the corresponding concepts and properties have been instantiated).

Finally, the queries let the user know about the specific instances that satisfy or contradict a requirement during the V&V processes. If a requirement is not met, this mechanism can also provide recommendations. In this respect, recommended possible actions to take include (i) limiting the extent to which the ATV is valid by refining or adding some requirements, or as a last resort modify the ATV, or (ii) improving the similarity between both kinds of systems, e.g., by modifying the LabSys or by suggesting changes to the industrial case (e.g., adding some sensors). This process is strongly supported by both the reasoning mechanism and the expressiveness of the querying language. Requirement fulfillment metrics are finally proposed, in order to assess the scalability of the ATV (for the V&V scope) from the LabSys to the IndSys considered.

The steps of the WONKA V&V processes of the ATV are presented in Figure 1.

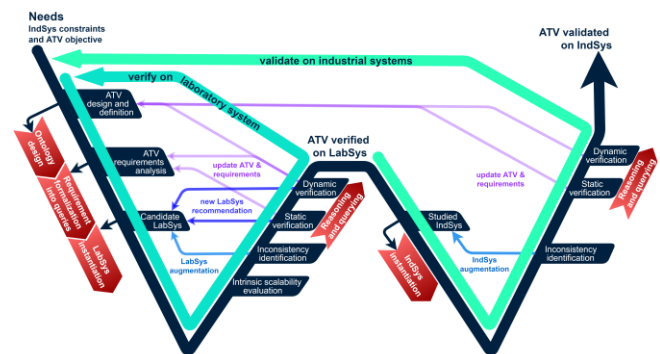


Figure 1: The WONKA framework for an ATV V&V process.

### 3.3 WONKA Case Study Description

The case study onto which this V&V methodology has been applied comprises a laboratory system, i.e. LabSys (coffee machine) and two industrial manufacturing systems, i.e. IndSys (HVAC and food manufacturing production line). The operating modes identification approach (BSS technique) to be validated, i.e. ATV, consists in a two-step process: using only a sensor-originated multivariate time series, the experiment should (i) use clustering to provide a sequence of the distinct system operating modes; the algorithm then (ii) decomposes the sequence of identified operating modes into a sequence of actuators activated during these operating modes (i.e., providing the actuators' activation states over time).

The laboratory use case consists of an automatic coffee machine with ground and bean coffee, made up of a pump, a heating coil, and a motor (to grind the beans). This LabSys was initially chosen for its compactness and the numerous energy transfers it contains. It was additionally instrumented with accelerometers, and later with current sensors.

The first IndSys is an industrial HVAC that regulates the temperature, humidity and air flow within a vehicle testbed. The actuators are divided in two groups: a heating coil, a cooling coil, a dehumidifier, and a fan which regulate the air properties arriving to the testbed; the air is then further cooled down and humidified inside the chamber. We restrict this study to the thermal perspective, where the actuators are mounted in series and controlled according to two temperature sensors.

The second IndSys considered is a mixing, refining and conching production line, as part of a chocolate factory. Only one machine with multiple tasks is considered for the purpose of this study. Three current sensors measure the machine's electricity consumption on all three phases supplying it.

The three use cases, i.e., the LabSys and the two IndSys, are heterogeneous in terms of energy domains at play, actuator architectures, and missions. Their interaction with their environment also differs. This intrinsic heterogeneity will be quantitatively evaluated during the WONKA process (as a semantic similarity between systems).

## 4 Laboratory System Selection and Augmentation Through ATV Verification

In order for the WONKA methodology to be effective (i.e., to cope with industrial system unavailability and limit experimentation costs), a laboratory system is selected and augmented according to three crucial steps, as per Figure 1. That is, the ontology is fine-tuned so as to represent all systems and translate requirements into compatible queries. Intrinsic scalability can thus be evaluated using semantic similarity metrics between laboratory and industrial systems. The static and dynamic verification steps assess a system's compliance with the requirements. Laboratory systems can be augmented so as to increase similarity and verification performance.

### 4.1 Needs Analysis

All requirements are determined using a traditional MBSE analysis, whether these concern the ATV, the systems themselves, their respective environments, or test scenarios. Each system with its environment can have its own set of requirements (constraints), independently from the ATV. Keeping all requirements in the final representation is important, especially when there exists cross-requirements or in the event of a disturbing effect of the ATV on the system. Then a set of requirements is selected in order to express how the ATV can be considered as verified or validated regarding the industrial issue addressed.

An MBSE method is used to elicit all ATV-related requirements. Regarding the application case study, we perform the black box analysis of the SE-READ (System Engineering based Requirements Elicitation & Architecture Design) methodology [30]. We define the requirements the studied systems (LabSys and IndSys) have to fulfill to verify and then to validate the ATV.

This methodology adopts an external point of view of the system of interest (SoI), i.e. here the ATV. This black box analysis supports the emergence of a comprehensive and consistent set of requirements. Based on different views, with potential iterations between them, this aims to answer the question: "what should my system do and with which external actors/systems?"

The case study focuses on the BSS ATV that meets the industrial issue of identifying the energy consumption of manufacturing systems regarding their operating modes (and their corresponding activated actuators) in an unsupervised way. It also considers the two industrial use cases (HVAC and chocolate production line). Based on these considerations, the first SE-READ view defines the requirement related to the main mission of the SoI, i.e. the ATV's mission: "*The ATV is a blind source separation method that shall identify and differentiate the operating modes of a black box system and decompose them in active actuators*". Then, the context view presents the SoI interactions with external systems and actors. In our case, this view allows to derive requirements from each IndSys and its environment, as external interactors of the BSS ATV. For example, an IndSys-derived requirement for this ATV is "*Any disturbance of the environment shall be measured*". Similarly, for the HVAC IndSys, these context-derived requirements relate to the HVAC capabilities. The HVAC system has to monitor and control the thermal environment within a specific temperature range, thus achieving the vehicle testbed's mission. The "Services provided by the SoI" view describes the different BSS ATV objectives, in relation with the ATV mission. This view allows to deduce some requirements inherent to the ATV. The final "Functional scenarios" view of the SE-READ black box analysis provides the requirements related to the description of the test sequences for experiments that have to verify dynamic requirements.

An extract of this MBSE analysis is summarized in Figure 2, where the LabSys (on the left) and the IndSys (on the right) both have to satisfy the ATV requirements, within the WONKA framework.

In order to qualitatively and quantitatively measure ATV requirements fulfillment, and identify precisely the elements that

do not satisfy such requirements (leading to some recommendations), we have to convert these requirements into questions. These questions will later be formalized into queries, to be supported by the ontological reasoning process. For the illustration, some selected requirements (in pink) have been translated into questions and presented in Table 1. This represents only a subset of the requirements to be validated on all use cases (LabSys & IndSys in their environment).

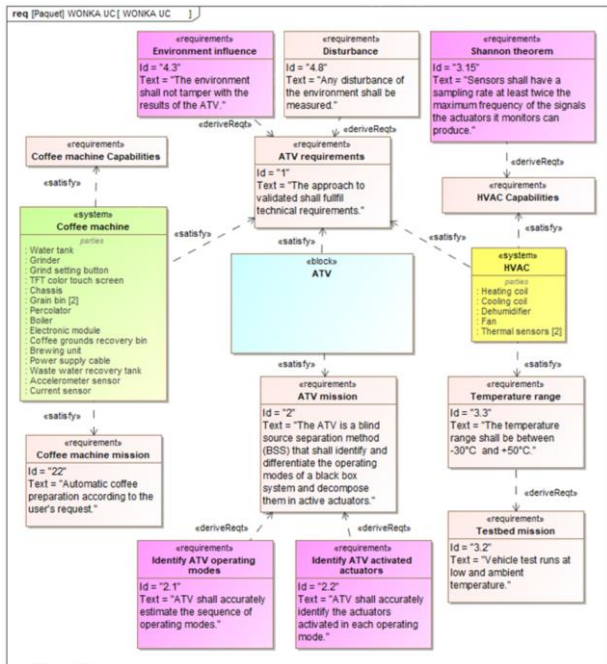


Figure 2: Requirements (extract) resulting from the MBSE analysis.

Requirement	Related question for query
<b>Req 3.15 (static)</b>	Does each sensor have a sampling rate at least twice the maximum frequency of the signals the actuators it monitors can produce?
<b>Req 2.1 (dynamic)</b>	Does the ATV accurately estimate the sequence of operating modes, given a performance threshold?
<b>Req 2.2 (dynamic)</b>	Does the ATV accurately identify the actuators activated in each operating mode?
<b>Req 2.7 (static)</b>	For each actuator within a system of interest, is there at least one sensor monitoring an observable property this actuator affects?
<b>Req 2.9 (static)</b>	In a sensor energy domain, are the actuator-originated signals coupled with one another? (i.e., actuators in series association)
<b>Req 4.3 (static)</b>	What are the elements of the environment that may tamper with the results of the ATV?

Table 1: Some selected requirements expressed as questions for querying the knowledge graph.

## 4.2 Ontology Design and Requirements Formalization

4.2.1 *Ontology design for describing the ATV and the use cases.* An ontology is a specification of a conceptualization [18]. It models knowledge as a collection of primary concepts and relationships between them, namely classes, object, and data properties in Web Ontology Language (OWL). Two main techniques are generally used to build an ontology: alignment to existing ontologies or appropriate ontology specification. Once the ontology is built, the systems and applications of interest can be formally represented. The instances (individuals), including their properties and values, must therefore be collected, and their relationships between one another be declared. These axioms form the knowledge base (KB). Ideally, such an ontology should be generic-enough to fit any system or application. Due to the difficulty of perfectly designing a domain ontology for heterogeneous systems (by opposition to task and application ontologies), enhancing its design after each iteration of the proposed methodology is preferred. Indeed, in research activities, phenomena are often better understood after series of experiments. Expressivity can also be increased by concept specialization. The use cases (systems and their environments) are then instantiated accordingly. As the ATV has to be applicable to both kinds of systems, laboratory and industrial, a single ATV instance will be used for both.

The methodology presented in this paper was implemented on an ontology inherited from SAREF and SAREF4INMA, deemed lightweight and sufficient for the purpose of this study. This framework allows to represent both industrial use cases and less conventional cyber-physical systems in laboratory environments. A typical implementation scheme is presented in Figure 3 as an RDF graph. This ontology was designed in Protégé [33].

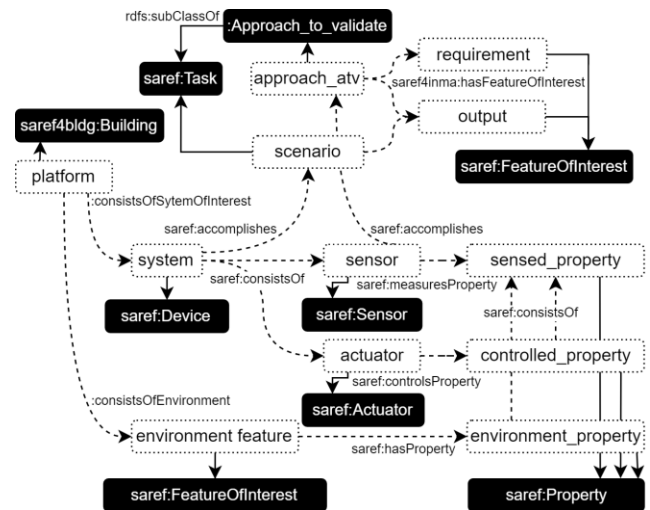


Figure 3: Implementation scheme of the knowledge base.

A platform, be it a laboratory or a plant/industrial system, consists of a system of interest (consistsOfSystemOfInterest),

an environment with miscellaneous features (consistsOfEnvironment), and possibly additional pieces of equipment (consistsOfAdditionalSystem). These relations are sub-properties of the consistsOf object property. This differentiation eases requirements formalization. Each Device or FeatureOfInterest relates to a Property, through hasProperty, controlsProperty or measuresProperty relationships. The properties themselves can relate to each other (consistsOf). This accounts for potential couplings in the systems' dynamics and the signals they originate (controlsProperty) or measure (measuresProperty). The system of interest serves a purpose through a dedicated scenario (Task).

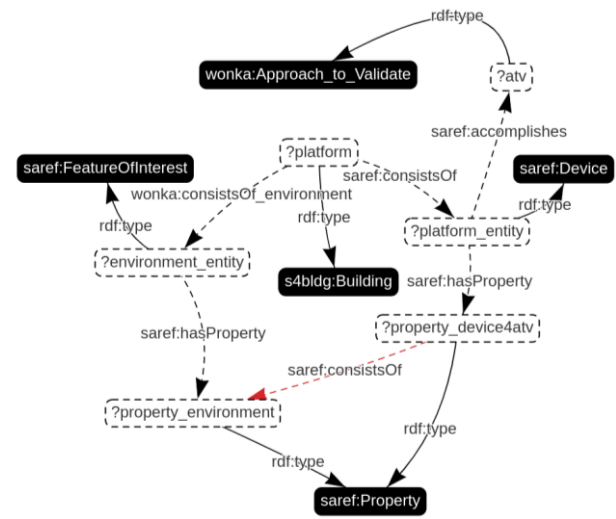
The ATV considered here, defined in our ontology as a subclass of a Task, uses exclusively sensor data. Hence every Sensor available accomplishes the ATV, i.e., provides input data to the ATV. The scenario and the ATV share an Output which may relate to a specific Requirement. In this context, an experiment's output consists of estimation accuracies (Req 2.1 and Req 2.2 respectively). Each of these requirements holds a validation threshold. Alternatively, this parameter can be held by the corresponding query. The performance of the process identification technique (operation and actuator-wise) can thus be evaluated. This output will therefore require an experiment to be computed. In order to compare each output to its corresponding requirement, they both relate to each other through the symmetric object property sameSettingAs, taken from the SSN ontology [20]. In general, should there be collections of instances of the same concepts but representing different aspects, they can be distinguished inside SPARQL queries using this object property. Overall, the knowledge base is composed of all three test cases; the ATV is uniquely defined and shared among them.

Only slight modifications were made to the SAREF ontology. The object properties hasProperty and consistsOf were made transitive. The domain of the former was also extended to the Device class. Relationships measuresProperty and controlsProperty were set as sub-properties of hasProperty. Thanks to these changes, all couplings between observable properties can be revealed after reasoning and associated to specific devices. As an example, in the HVAC use case, despite the presence of five actuators between the outside temperature and the temperature measured by the first sensor, query Req 4.3 is capable of detecting an effect of the environment on the ATV (the actuator-originated properties act as proxys in this case). Inference is finally performed using the HermitT reasoner [16].

**4.2.2 Requirements formalization into queries.** We distinguish two types of requirements. The static ones depend only on use case definition (system architecture, environment, tasks, etc.), whereas the dynamic requirements must be evaluated through experiments. The requirements must be formatted in an ontological format, i.e., as a formal statement, in order to be checked by the querying engine during the V&V process. The Resource Description Framework (RDF) is therefore used. The requirements are more

specifically translated into queries using the Simple Protocol And RDF Query Language (SPARQL). Not only should the requirements have a SPARQL query selecting the elements that will satisfy them; they should also include a query for the non-validating elements. To avoid any loss of genericity, requirements formalization is restricted to the domain ontology, i.e., the queries should not contain specializations of the core concepts, so as to be applicable to all use cases. For this reason, WONKA forces the user to conceptualize the validation criteria and better capture the limits of the ATV.

Once the ontology has been designed, each requirement is formally expressed as a SPARQL query. Such a query can be parsed and visually represented as an RDF graph, as illustrated in Figure 4 with Req 4.3.



**Figure 4: Requirement Req 4.3 represented as an RDF graph (in red, the relationship must not exist).**

This requirement seeks the individuals (environment features) that may cause the ATV to fail or underperform. In the WONKA framework, this requirement is expressed as follows: a platform (use case) consists of an entity including the environment (e.g., weather) and a device (e.g., sensor) – accomplishing the ATV –, each having a property (e.g., outside and sensed temperatures respectively). The property of the device must not consist of (i.e., be affected by) that of the environment. Using parent concepts (e.g., using Device and FeatureOfInterest concepts instead of TemperatureSensor and Weather, the latter being subclasses of the former) allows to cover most interactions between a system and its environment.

The ontology acts as a semantic embedding: in addition to making the (formally defined) requirements evaluable for all use cases, the solutions found by WONKA can have different meanings depending on the use case. For instance, the vehicle (acting as a temperature disturbance in the testbed) and the coffee machine's power supply are both subclasses of FeatureOfInterest in the upper ontology (SAREF, essentially), despite having very different natures.

### 4.3 Laboratory System Analysis and Selection

Once the ontology is designed, the knowledge graph and the queries are established, our procedure considers three activities in order to select a LabSys that verifies the ATV: (i) assess the extent to which each laboratory system is representative of the industrial ones, (ii) augment the laboratory system so as to increase verification indicators, (iii) verify the ATV on the selected laboratory system, based on the industrial use case requirements.

*4.3.1 System analysis and intrinsic scalability evaluation.* Before launching expensive or time-consuming experiments, candidate laboratory systems can be prioritized using their *intrinsic scalability*. A scale factor and a similarity index between a LabSys and an IndSys are therefore computed. This scalability is intrinsic in that use cases are compared semantically, leaving out their connections to the ATV considered and related requirements. First, the respective subgraphs of the use cases are extracted from the knowledge graph. Every element of a use case originates from a single platform (a `Building` or one of its subclasses, as shown in Figure 3), and use cases are connected solely via the ATV. Subgraph extraction is thus straightforward. Let *Lab* and *Ind* be the sets of individuals in the LabSys and IndSys respectively. A simple scale factor is given as the ratio of *Lab*'s cardinality to *Ind*'s. Similarly to [31], we propose the following (symmetric) similarity metric between two sets of individuals.

$$sim(Lab, Ind) = \frac{1}{2} \left( \frac{\sum_{w_{lab} \in Lab} maxSim(w_{lab}, Ind)}{|Lab|} + \frac{\sum_{w_{ind} \in Ind} maxSim(w_{ind}, Lab)}{|Ind|} \right) \quad (1)$$

where  $|.$  is the cardinality operator,  $w_{lab}$  and  $w_{ind}$  are individuals. *maxSim* denotes the highest similarity index between the class of the first term and each distinct class represented in the set of individuals in the second term. There exist multiple metrics tailored to assess the similarity between two concepts [19]. Any of these can be used, yet care must be taken that the two concepts considered refer to different knowledge subgraphs (despite belonging to the same ontology). The semantic similarity metric introduced by Lin in [28] was adapted as:

$$sim_{lin}(c_1, c_2) = \frac{IC_{KG1}(LCS(c_1, c_2)) + IC_{KG2}(LCS(c_1, c_2))}{IC_{KG1}(c_1) + IC_{KG2}(c_2)} \quad (2)$$

where the least common subsumer (LCS) is the most specific common ancestor to a pair of concepts, and  $IC_{KG}$  denotes the information content (IC) [28] computed within a knowledge graph  $KG$ . In this context,  $KG1$  and  $KG2$  refer to the knowledge subgraphs of the LabSys and the IndSys respectively.

Alternatively, edge-based similarity metrics can account for relationships, and thus evaluate to what extent components are similarly interconnected.

Originally, three laboratory systems were considered: a 6-axis robotic arm, a 3-axis Fused Deposition Modelling (FDM) 3D printer, and an automatic bean-to-cup coffee machine.

Due to page restrictions, these systems are not presented here, yet their scalability (scale factor  $r$ ) and similarity metrics are summarized in Table 2.

IndSys \ LabSys	Robotic arm	3D printer	Coffee machine
Vehicle testbed	$r = 1:1.35$ 81.2%	$r = 1:1.35$ 81.1%	$r = 1:1.40$ 83.4%
Chocolate factory	$r = 1:1.31$ 83.2%	$r = 1:1.31$ 97.0%	$r = 1:1.36$ 98.2%

**Table 2: Intrinsic scalability of different LabSys applied to the case studies.**

The systems are represented in the same ontology, consistently with the ATV considered. This explains the high similarities observed in the plane of analysis. It is worth noting that actuators and physical properties total up to 50% of the instances of a system. Similar energy domains and actuators thus result in greater similarities. Interestingly, the coffee machine (augmented as per section 4.3.2) is the best candidate at this stage.

*4.3.2 Laboratory system augmentation.* The ATV, LabSys or IndSys can be adapted according to WONKA recommendations. Indeed, the querying mechanism considered in this paper allows to pinpoint specific instances unable to meet a requirement. Suitable additions according to these abstract recommendations include supplementary components or relationships. For instance, changing a sensor's location may allow it to monitor another machine. This will result in better verification performance and may increase the scalability index. Laboratory system augmentation can be performed iteratively by updating the knowledge base and assessing scalability and verification performances.

The recommendation mechanism is illustrated in Figure 5. The validating query of Req 2.7, ensuring that all actuators can be monitored by a sensor, is instantiated. WONKA indicates that no sensor, observed property nor actuator-induced property were able to fulfill this requirement. Nodes `?sensor`, `?sensprop`, `?actprop` denote variables. From the query, their respective class is known. The resulting recommendation is to add a sensor within the laboratory, measuring an observable property induced by an actuatable property of the heating coil.



**Figure 5: Instantiated query (Req 2.7) with modification proposals to validate the corresponding requirement (green is valid, red is missing, not instantiated and to be added).**



**4.3.3 Laboratory system selection based on verification.** This step addresses the need to identify a LabSys that fulfills static and dynamic requirements related to the proposed ATV. Once the knowledge graph is deemed complete, queries can be applied. Based on the results of these queries, the degree of verification or validation is simply given as the ratio of the number of verified requirements to the total number of requirements. It is also possible to use the number of combinations of individuals (i.e., each row in the SPARQL query results) satisfying the requirements over the total number of combinations of individuals concerned by those requirements. Both indicators are useful to the analysis. The former weighs the requirements equally. This verification status indicates whether the ATV will be verified on a use case. Whereas the latter allows to pinpoint the most unsuccessfully verified requirements and analyze them in more detail.

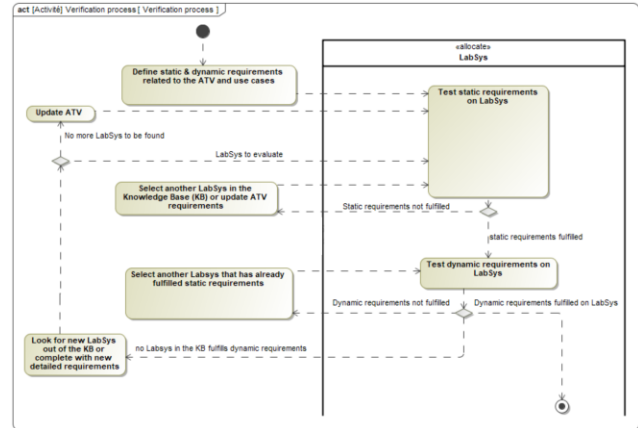
The degree of verification is first computed with respect to static requirements. The LabSys achieving the highest score is selected to conduct experiments. As experiments are conducted (i.e., the realization of a scenario), additional constraints may arise, suggesting to complete the ATV requirements. This makes the verification process more robust and reliable. These constraints can be incorporated to the set of static or dynamic requirements. The degree of verification with respect to the ATV dynamic requirements is then evaluated.

In the example developed in the case study, a new requirement on the noise level of the sensors arose from LabSys experiments, as the BSS algorithm proved unsuccessful when using low-cost sensors.

Initially, we considered only the requirements Req 3.15, 2.9, and 4.3, as described in section 4.1 and in line with the needs of the ATV. Req 3.15 relates to system capability. The constraints Req 2.9 and 4.3 stem from an analysis of the source separation technique being used in the ATV. Clustering is performed to separate distinct operations; a subset of the centroids is selected to form a basis onto which all centroids are projected. Consequently, coupled dynamics between actuators is undesirable (Req 2.9), as the projection would no longer represent the true actuator combinations. No to moderate influence of the environment (Req 4.3) is also necessary to prevent environmental phenomena from being detected as actuators in this BSS problem (i.e., actuators identification). Accelerometers being non-intrusive and widely used in industrial applications, we first equipped the coffee machine with such sensors upon WONKA recommendations. As the static requirements had been fulfilled for the LabSys, an experiment was run upon the LabSys. This experiment consists in recording sensor data while making a coffee.

As a result of this experiment, the dynamic requirements Req 2.1 and Req 2.2 were not met, due to the lack of vibrations the heating operation emits. The set of requirements was hence completed with the addition of Req 2.7. Consequently, the LabSys was equipped with a current sensor located on the power supply. Another experiment then proved successful. The knowledge base was updated accordingly. At this stage, all the requirements considered have been met, thus verifying the ATV on the LabSys.

For conciseness, the verification process on the laboratory system is summarized in an activity diagram in SysML in **Figure 6**.



**Figure 6: The verification process in WONKA.**

## 5 Validation of the Scientific Approach

At this stage, there exist a LabSys onto which the ATV is verified, both statically and dynamically. Though the verification process, the ATV was further specified. Some queries are soft constraints rather than strict requirements. The resulting set of requirements and associated queries is used to validate the ATV on the industrial use cases. Applying the static queries to the industrial systems allows to anticipate the extent to which the ATV is likely to be validated on an IndSys. The V&V performance metrics defined in section 4.3.3 are computed. Depending on the nature of the results, IndSys augmentation may be necessary. Furthermore, unforeseen constraints or limits may arise, leading to a partial re-design of the ATV. In particular, when some queries related to dynamic requirements are not verified by the ATV on the studied systems, the framework will recommend to update ATV.

On the case study, all requirements on the LabSys were completely verified. Yet the application of the ATV onto the two IndSys could not even pass the static requirements. A lightweight requirement analyzer was developed with a view to locate individuals and object properties that do not meet the requirements, i.e., missing relationships or erroneous ones. This analysis algorithm uses Owlready2 [26] for knowledge base manipulation and RDFLib<sup>1</sup> for parsing the SPARQL queries. The validity of an edge within the knowledge graph, or its undesired absence, is obtained by comparing the validating and non-validating requirements-related queries. Their RDF graphs are instantiated, and the result is fed into the complete knowledge graph. Overall, instances and relationships that meet the requirement are colored differently than missing or incorrect elements, the latter having a higher priority. This visual aid eases the comparison between use cases in the ontology plane, from an abstract perspective.

<sup>1</sup> <https://github.com/RDFLib>

At last, using the validation metrics earlier defined, the chocolate factory use case shows a 75% validation rate (96% in terms of validating combinations of instances) on the static requirements, against 50% for the vehicle testbed (68% in terms of validating combinations of instances). This is consistent with the results observed during the experiments on both IndSys: the ATV performed better on the chocolate factory than it did on the testbed. Thanks to these metrics, we were able to easily determine the problematic requirements and components for validating the ATV on the industrial systems under consideration. Actuators' dynamics appearing as coupled from a sensor's perspective is the primary cause of ATV failure when applied to a use case. This occurs multiple times in the HVAC use case, where the actuators are in series association. This prevents the ATV from dissociating them correctly. The second plausible cause is the effect of the environment on the ATV, as both the chocolate manufacturing machine and the HVAC are affected by external disturbances (room temperature and outside air temperature respectively). In the chocolate factory, room temperature influences the chocolate's viscosity which leads to a disturbed load torque and current consumption. However, as anticipated with the validation of most requirements, the ATV correctly identified the operations and actuators in an unsupervised fashion. The validation process has been summarized in an activity diagram in SysML in Figure 7.

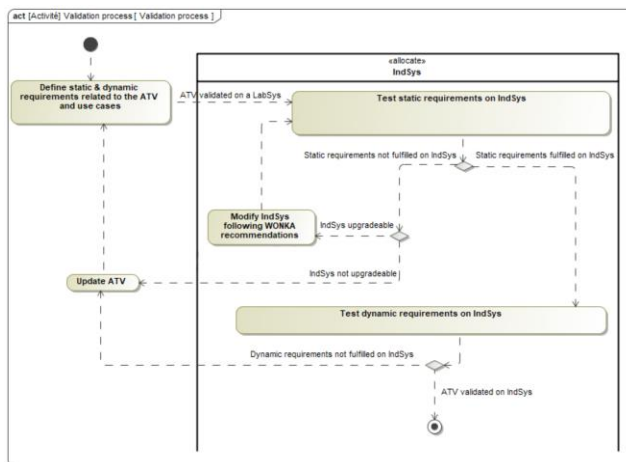


Figure 7: The validation process in WONKA.

## 6 Conclusions

In this paper, the WONKA framework was proposed, together with a generic V&V process, supporting MBSE activities, for pre-validating a scientific approach on a laboratory system before being applied to an industrial use case. By formally describing systems, applications, and V&V concepts (the requirements in particular), this methodology thus enables scalability assessment between heterogeneous systems.

This methodology was applied to the abovementioned case study. The study showed the BSS algorithm considered turns out not to be applicable to HVAC use cases due to unforeseen limits. The ATV was successfully verified on the coffee machine (instrumented with a current sensor) and validated (to some extent) on the chocolate factory use case.

Future work will primarily focus on better assessing this scalability. In particular, the use of density and topology metrics will be investigated, so as to locate and evaluate structural differences and similarities between systems and proposed solutions.

The proposed methodology was mainly developed in an effort to cope with system unavailability. The WONKA framework will include cost and time effectiveness aspects, both with respect to the experiments necessary to the dynamic V&V and system upgrade (LabSys or IndSys augmentation).

The perimeter of WONKA will also be extended to the human factor. Indeed, this study was limited to physical components and systems (hardware or software). The future modules (e.g., similarly to the FBS [11] or DUL [20] ontologies) of the WONKA ontology will need to tackle the impact of the recommended infrastructure modifications (addition of sensors, etc.) on operators, associated skills, maintenance, and any non-physical modifications.

## ACKNOWLEDGMENTS

EnerMan has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 958478.

## REFERENCES

- [1] Issa Atoum. 2019. *A Scalable Operational Framework for Requirements Validation Using Semantic and Functional Models*. DOI:https://doi.org/10.1145/3305160.3305166
- [2] Franz Baader. 2014. *Ontology-Based Monitoring of Dynamic Systems*. In *Fourteenth International Conference on the Principles of Knowledge Representation and Reasoning*. Retrieved January 13, 2022 from https://www.aaai.org/ocs/index.php/KR/KR14/paper/view/7972
- [3] Brian Bailey. 2007. *Address verification issues with scalable methods*. (2007), 5.
- [4] Jason Baumgartner, Hari Mony, Viresh Paruthi, Robert Kanzelman, and Geert Janssen. Scalable sequential equivalence checking across arbitrary design transformations. In *Proc. ICCD '07*, 259–266.
- [5] Frédéric Bénében, Chihab Hanachi, Matthieu Luras, Pierre Couget, and Vincent Chapurlat. 2008. *A metamodel and its ontology to guide crisis characterization and its collaborative management*. In *Proceedings of the 5th International Conference on Information Systems for Crisis Response and Management (ISCRAM)*, Washington, DC, USA, May, 4–7.
- [6] Devesh Bhatt, Gabor Madl, David Oglesby, and Kirk Schloegel. 2010. *Towards Scalable Verification of Commercial Avionics Software*. In *AIAA Infotech@Aerospace 2010*, American Institute of Aeronautics and Astronautics, Atlanta, Georgia. DOI:https://doi.org/10.2514/6.2010-3452
- [7] Milton Borsato. 2014. *Bridging the gap between product lifecycle management and sustainability in manufacturing through ontology building*. *Computers in Industry* 65, 2 (February 2014), 258–269. DOI:https://doi.org/10.1016/j.compind.2013.11.003
- [8] Qiushi Cao, Cecilia Zanni-Merk, and Christoph Reich. 2018. *Ontologies for Manufacturing Process Modeling: A Survey*.
- [9] Michael Compton, Payam Barnaghi, Luis Bermudez, Raúl García-Castro, Oscar Corcho, Simon Cox, John Graybeal, Manfred Hauswirth, Cory Henson, Arthur Herzog, Vincent Huang, Krzysztof Janowicz, W. David Kelsey, Danh Le Phuoc, Laurent Lefort, Myriam Leggieri, Holger Neuhaus, Andriy

- Nikolov, Kevin Page, Alexandre Passant, Amit Sheth, and Kerry Taylor. 2012. The SSN ontology of the W3C semantic sensor network incubator group. *Journal of Web Semantics* 17, (December 2012), 25–32. DOI:<https://doi.org/10.1016/j.websem.2012.05.003>
- [10] Javier Cuenca, Felix Larrinaga, and Edward Curry. A Unified Semantic Ontology for Energy Management Applications. 12.
- [11] Jean-Yves Dantan, Ismail El Mouayni, Leyla Sadeghi, Ali Siadat, and Alain Etienne. 2019. Human factors integration in manufacturing systems design using function-behavior-structure framework and behaviour simulations. *CIRP Annals* 68, 1 (January 2019), 125–128. DOI:<https://doi.org/10.1016/j.cirp.2019.04.040>
- [12] Romain Delabeye, Martin Ghienne, Arkadiusz Kosecki, and Jean-Luc Dion. 2022. Unsupervised manufacturing process identification using non-intrusive sensors. Retrieved September 7, 2022 from <https://hal.archives-ouvertes.fr/hal-03768559>
- [13] P. E. Farrell, M. D. Piggott, G. J. Gorman, D. A. Ham, C. R. Wilson, and T. M. Bond. 2011. Automated continuous verification for numerical simulation. *Geoscientific Model Development* 4, 2 (2011), 435–449.
- [14] Stefano Gagliardo, Franca Giannini, Marina Monti, Giulia Pedrielli, Walter Terkaj, Marco Sacco, Matteo Ghellere, and Francesco Salamone. 2015. An Ontology-based Framework for Sustainable Factories. *Computer-Aided Design and Applications* 12, 2 (March 2015), 198–207. DOI:<https://doi.org/10.1080/16864360.2014.962432>
- [15] Antonio Giovannini, Alexis Aubry, Hervé Panetto, Michele Dassisti, and Hind El Haouzi. 2012. Ontology-based system for supporting manufacturing sustainability. *Annual Reviews in Control* 36, 2 (2012), 309–317.
- [16] Birte Glimm, Ian Horrocks, Boris Motik, Giorgos Stoilos, and Zhe Wang. 2014. HermiT: An OWL 2 Reasoner. *J Autom Reasoning* 53, 3 (October 2014), 245–269. DOI:<https://doi.org/10.1007/s10817-014-9305-1>
- [17] D. Groen, H. Arabnejad, V. Jancauskas, W. N. Edeling, F. Jansson, R. A. Richardson, J. Lakhilili, L. Veen, B. Bosak, P. Kopta, D. W. Wright, N. Monnier, P. Karlsruhofer, D. Suleimenova, R. Sinclair, M. Vassaux, A. Nikishova, M. Bieniek, Onnie O. Luk, M. Kulczewski, E. Raffin, D. Crommelin, O. Hoenen, D. P. Coster, T. Piontek, and P. V. Coveney. 2021. VECMAtk: a scalable verification, validation and uncertainty quantification toolkit for scientific simulations. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 379, 2197 (May 2021), 20200221. DOI:<https://doi.org/10.1098/rsta.2020.0221>
- [18] Thomas R. Gruber. 1993. Stanford University. Toward Principles for the Design of Ontologies Used for Knowledge Sharing.
- [19] Mohamed Ali Hadj Taieb, Mohamed Ben Aouicha, and Abdelmajid Ben Hamadou. 2014. Ontology-based approach for measuring semantic similarity. *Engineering Applications of Artificial Intelligence* 36, (November 2014), 238–261. DOI:<https://doi.org/10.1016/j.engappai.2014.07.015>
- [20] Armin Haller, Krzysztof Janowicz, Simon Cox, Maxime Lefrançois, Kerry Taylor, Danh Le Phuoc, Joshua Lieberman, Raúl García-Castro, Rob Atkinson, and Claus Stadler. 2018. The Modular SSN Ontology: A Joint W3C and OGC Standard Specifying the Semantics of Sensors, Observations, Sampling, and Actuation. *Semantic Web – Interoperability, Usability, Applicability* (2018). Retrieved May 14, 2022 from <https://hal.archives-ouvertes.fr/hal-01885335>
- [21] Mark D. Hill. 1990. What is scalability? *SIGARCH Comput. Archit. News* 18, 4 (décembre 1990), 18–21. DOI:<https://doi.org/10.1145/121973.121975>
- [22] IEEE Std 1012. 2016. *IEEE Standard for System, Software, and Hardware Verification and Validation*. IEEE. DOI:<https://doi.org/10.1109/IEEESTD.2017.8055462>
- [23] ISO/IEC/IEEE 1528. 2002. ISO/IEC/IEEE 15288 - SEBoK. Retrieved January 5, 2020 from [https://www.sebokwiki.org/wiki/ISO/IEC/IEEE\\_15288](https://www.sebokwiki.org/wiki/ISO/IEC/IEEE_15288)
- [24] Krzysztof Janowicz, Armin Haller, Simon J. D. Cox, Danh Le Phuoc, and Maxime Lefrançois. 2019. SOSA: A lightweight ontology for sensors, observations, samples, and actuators. *Journal of Web Semantics* 56, (May 2019), 1–10. DOI:<https://doi.org/10.1016/j.websem.2018.06.003>
- [25] Mounira Kezadri and Marc Pantel. 2012. First steps toward a Verification and Validation Ontology. In *Embedded Real Time Software and Systems (ERTS2012)*, Toulouse, France. Retrieved January 7, 2022 from <https://hal.archives-ouvertes.fr/hal-02189894>
- [26] Jean-Baptiste Lamy. 2017. Owlready: Ontology-oriented programming in Python with automatic classification and high level constructs for biomedical ontologies. *Artificial Intelligence in Medicine* 80, (July 2017), 11–28. DOI:<https://doi.org/10.1016/j.artmed.2017.07.002>
- [27] Severin Lemaignan, Ali Siadat, J.-Y. Dantan, and Anatoli Semenenko. 2006. MASON: A proposal for an ontology of manufacturing domain. In *IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06)*, IEEE, 195–200.
- [28] Dingquan Li. An Information-Theoretic Definition of Similarity. 44.
- [29] David Long. 2021. Schema, Metamodels, and Ontologies, Oh My! In <https://www.youtube.com/watch?v=47qCVeLiUE>. Annual INCOSE International workshop. Retrieved from [https://www.omgwiki.org/MBSE/lib/exe/fetch.php?media=mbse:incose\\_mbse\\_iw\\_2021:schemametamodelontology\\_long.pdf](https://www.omgwiki.org/MBSE/lib/exe/fetch.php?media=mbse:incose_mbse_iw_2021:schemametamodelontology_long.pdf)
- [30] Faïda Mhenni, Jean-Yves Choley, Olivia Penas, Régis Plateaux, and Moncef Hammadi. 2014. A SysML-based methodology for mechatronic systems architectural design. *Advanced Engineering Informatics* 28, 3 (August 2014), 218–231. DOI:<https://doi.org/10.1016/j.aei.2014.03.006>
- [31] Rada Mihalcea. Corpus-based and Knowledge-based Measures of Text Semantic Similarity. 6.
- [32] Georgios M. Milis, Christos G. Panayiotou, and Marios M. Polycarpou. 2019. SEMIoTICS: Semantically Enhanced IoT-Enabled Intelligent Control Systems. *IEEE Internet of Things Journal* 6, 1 (February 2019), 1257–1266. DOI:<https://doi.org/10.1109/JIOT.2017.2773200>
- [33] Mark A. Musen. 2015. The Protégé Project: A Look Back and a Look Forward. *AI Matters* 1, 4 (June 2015), 4–12. DOI:<https://doi.org/10.1145/2757001.2757003>
- [34] David Lira Nuñez and Milton Borsato. 2018. OntoProg: An ontology-based model for implementing Prognostics Health Management in mechanical machines. *Advanced Engineering Informatics* 38, (October 2018), 746–759. DOI:<https://doi.org/10.1016/j.aei.2018.10.006>
- [35] Hervé Panetto, Michele Dassisti, and Angela Tursi. 2012. ONTO-PDM: Product-driven ONTOlogy for Product Data Management interoperability within manufacturing process environment. *Advanced Engineering Informatics* 26, 2 (2012), 334–348.
- [36] Amine Raji and Philippe Dhaussy. 2011. Use Cases Modeling for Scalable Model-Checking. In *2011 18th Asia-Pacific Software Engineering Conference*, 65–72. DOI:<https://doi.org/10.1109/APSEC.2011.25>
- [37] Victor Julio Ramírez-Durán, Idoia Berges, and Arantza Illarramendi. 2020. ExtrOnt: An ontology for describing a type of manufacturing machine for Industry 4.0 systems. *Semantic Web* 11, 6 (2020), 887–909.
- [38] Maria Ines Robles, Edgar Ramos, Nicklas Beijar, and Nanjangud C. Narendra. 2017. Calculating LWM2M resource semantic distance through SENACT ontology. In *Proceedings of the Seventh International Conference on the Internet of Things (IoT '17)*, Association for Computing Machinery, New York, NY, USA, 1–8. DOI:<https://doi.org/10.1145/3131542.3131555>
- [39] Mike de Roode, Alba Fernández-Izquierdo, Laura Daniele, María Poveda-Villalón, and Raúl García-Castro. 2020. SAREF4INMA: A SAREF extension for the industry and manufacturing domain. *Semantic Web* 11, 6 (January 2020), 911–926. DOI:<https://doi.org/10.3233/SW-200402>
- [40] Mike Ryan, Stephen Cook, and William R. Scott. 2013. Application of MBSE to requirements engineering-Research challenges. (2013).
- [41] Abdelkader Magdy Shaaban, Christoph Schmittner, Thomas Gruber, A. Baith Mohamed, Gerald Quirchmayr, and Erich Schikuta. 2019. Ontology-Based Model for Automotive Security Verification and Validation. In *Proceedings of the 21st International Conference on Information Integration and Web-based Applications & Services*, ACM, Munich Germany, 73–82. DOI:<https://doi.org/10.1145/3366030.3366070>
- [42] Dimitrios Tsoukalas, Miltiadis Siavvas, Maria Mathioudaki, and Dionysios Kehagias. 2021. An Ontology-based Approach for Automatic Specification, Verification, and Validation of Software Security Requirements: Preliminary Results. In *2021 IEEE 21st International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, 83–91. DOI:<https://doi.org/10.1109/QRS-C55045.2021.00022>
- [43] Michael Uschold and Michael Grüninger. 1996. Ontologies: Principles, methods and applications. *The Knowledge Engineering Review* 11, (January 1996).
- [44] Hendro Wicaksono, Fabian Jost, Sven Rogalski, and Jivka Ovtcharova. 2014. Energy Efficiency Evaluation in Manufacturing Through an Ontology-Represented Knowledge Base. *Intelligent Systems in Accounting, Finance and Management* 21, 1 (2014), 59–69. DOI:<https://doi.org/10.1002/isaf.1347>
- [45] Cheng Xu, Elisabeth Källström, Tore Risch, John Lindström, Lars Håkansson, and Jonas Larsson. 2017. Scalable validation of industrial equipment using a functional DSMS. *J Intell Inf Syst* 48, 3 (June 2017), 553–577. DOI:<https://doi.org/10.1007/s10844-016-0427-2>