MODELING A SUPPLY CHAIN REFERENCE ONTOLOGY BASED ON A TOP-LEVEL ONTOLOGY

Farhad Ameri
Associate Professor
Engineering Informatics Lab
Texas State University
San Marcos, Texas, USA
ameri@txstate.edu

Boonserm Kulvatunyou
Research Scientist
Systems Integration Division
National Institute of Standards and Technologies (NIST)
Gaithersburg, Maryland, USA
serm@nist.gov

ABSTRACT

Several supply-chain ontologies have been introduced in the past decade with the promise of enabling supply chain interoperability. However, the existing supply-chain ontologies have several gaps with respect to completeness, logical consistency, domain accuracy, and the development approach. In this work, we propose a new, supply-chain, reference ontology that is 1) based on an existing top-level ontology and 2) developed using a collaborative, ontology-development, best practice. We chose this approach because empirical studies have shown the usefulness of adopting a top-level ontology both for improving the efficiency of the development process and enhancing the quality of the resulting ontology. The proposed proof-of-concept reference ontology is developed in the context of the Industrial Ontology Foundry (IOF). IOF is an international effort aimed at providing a coherent set of modular and publicly-available ontologies for the manufacturing sector. Although the proposed reference ontology is still at the draft stage, this paper shows that it has already benefited from the collaborative development process that involves inputs from the other working groups within IOF. Additionally, as a way to validate the proposed reference ontology, an application ontology related to a supplier discovery and evaluation use case is derived from the reference ontology and tested.

Keywords: **supply chain, reference ontology, manufacturing, collaborative ontology development, interoperability**

INTRODUCTION

Supply chains are increasingly more complex, digital, and dynamic. In this context, supply-chain integration is a necessary feature to enable enhanced coordination and communication among various supply chain participants such as vendors, service providers, and customers [1]. Many supply chain researchers and practitioners support the idea that supply chain efficiency can be improved with seamless flow of information [2]. One of the main enablers of such a seamless flow of information is interoperability. Interoperability is the ability of two or more systems to exchange information and interpret the exchanged information meaningfully and accurately in order to produce useful results via deference to a common information exchange reference model [3].

To date, supply chain interoperability is still a major, unsolved problem. The existing supply chain solutions have not been able to achieve full or agile information integration, because they do not interoperate [4]. Lack of interoperability can be attributed to differences in the underlying semantics and business rules implemented by different supply chain software systems.

Ontologies have been proposed as the solution to these differences. Simply put, an ontology, which is a controlled vocabulary represented by formal logic, provides a consensus-based set of terms for describing the types of entities in a given domain and the relations between them [5]. In the supply chain domain, the core entities include the organizations that form the supply chain, their internal functions, capabilities, and resources, the buying and selling processes, the materials and the information that flow throughout the supply chains, and the processes and services that govern the operation and coordination of the supply chain.

The first and most basic benefit of an ontology is that, like other kinds of standard data models such as entity-relationship model and XML Schema, it provides a common terminology that can be used for data annotation [6]. This common terminology enables both machines and humans to access, understand, search, and retrieve data more efficiently.

A secondary benefit of ontology stems from its logic-based nature. Unlike other kinds of data models, logically formulated ontologies allow human and machine agents to make *inferences* about operations such as data aggregation, comparison, querying, and quality assurance. In addition, when data models are annotated or tagged by ontological entities, they become more easily searchable, combinable, and analyzable using logical-reasoning implemented by compatible software tools.

These benefits are the main reasons that researchers have been proposing a growing number of supply chain ontologies [7-10]. Grubic and Fan [11] studied the existing supply-chain ontologies; they concluded that those ontologies have failed to solve the current interoperability problems. The study identified several gaps in the existing supply chain ontology models [12]. Those gaps include weak methodological approaches, restricted and static views of supply chains, missing accounts of material traceability and service, and the dominance of taxonomies over formalized definitions.

A key conclusion from these studies is that "too much emphasis is placed on the organization and structure of human knowledge of supply chains rather than on understanding the ***reality*** of supply chains" (p.776). Ironically, the existence of these ontologies, which are based on varying and conflicting views of the supply-chain domain, has contributed to the interoperability problem rather than serving as a solution.

The objective of this current research is to investigate a method to develop a supply chain reference ontology based on a shared and domain independent, foundational ontology called Basic Formal Ontology (BFO). BFO's main difference from other top-level ontologies is its focus on reality (rather than an application or domain-specific view) and its past successes in using BFO in different domains. In the rest of the paper, we provide a background on BFO and on IOF where this research has been conducted. Then we outline the ontology development method. The draft supply chain reference ontology (SCRO) is presented, followed with a validation use case. Finally, the concluding remarks are provided.

INDUSTRIAL ONTOLOGIES FOUNDARY (IOF)

The IOF project is an international effort, with the participation of governments, industries, academia, and standard organizations. The vision of IOF is to make its ontologies publicly available and loyalty free in order to increase ontology adoptions in the manufacturing sector. The scope spans the entire domain of digital manufacturing in order to advance software and data interoperability.

The IOF results, once fully developed, will provide an open-source platform for developing, validating, aligning, sharing, and curating industrial ontologies. Rather than being an academic endeavor, IOF is committed to meet the needs of industrial stakeholders by providing reliable, turnkey solutions and by giving them best practices to integrate ontologies in their businesses. The technical goals of IOF include [12]:

- Create open, principles-based ontologies from which other domain-dependent or application-specific ontologies can be derived in a modular fashion.
- Ensure that IOF ontologies are non-proprietary and non-implementation-specific, so they can be reused in different industrial subdomains and standard bodies.
- Provide principles and best practices by which quality ontologies will support interoperability
- Institute a governance mechanism to maintain and promulgate the goals and principles.
- Provide an organizational framework and governance processes that ensure conformance to IOF principles and best practices.

Currently there are five active working groups (WGs) in IOF. Four of them addresses different subdomains of manufacturing including supply chain, production planning and scheduling, maintenance, and product-service systems. The last working group, namely the top-down WG, serves as the glue by providing a common ontology and ensuring the consistency across other working groups. The working groups receive support with respect to ontology development expertise from the members of the Technical Oversight Board (TOB) consisting of both ontology-inclined domain experts and ontologists [13]. Domain experts identify their interoperability requirements and ensure that the ontologists create definitions and axioms that meet those requirements.

Despite the broad scope of IOF, the ontologies it develops still must become a work item in existing standard development organizations (SDOs). One possible strategy is for IOF to develop detailed ontologies based on a few industry use cases. These detailed ontologies can then be modularized into mid-level reference ontologies and extended into domain-specific ontologies. IOF will maintain the mid-level reference ontologies; The SDOs will focus on the domain-specific ontologies There are two main concerns with this strategy: protecting all intellectual property rights and free access to the standards.

TOP-LEVEL ONTOLOGIES (TLO)

Ontologies can enable semantic interoperability when they are built according to a rigorous, multi-tiered, hierarchical architecture (Figure 1). Such an architecture has 1) a single, small, domain-neutral ontology at the top of this hierarchy and 2) a suite of lower-level ontologies – both domain-dependent and domain-specific ontologies. Top-level ontologies (a.k.a upper ontologies, or sometimes positioned in the opposite side as foundational ontologies) are highly abstract, domain-neutral because they establish a common framework for creating application ontologies [5].

An important function of an upper ontology is to support semantic interoperability by providing accurate and axiomatic definitions of the generic entities that can be further specialized by domain-specific ontologies. Some of the notable upper level ontologies include Basic Formal Ontology (BFO) [5], Domain Ontology for Linguistic and Cognitive Engineering (DOLCE) [14], PSL [15], and Suggested Upper Merged Ontology (SUMO) [16].

There are some differences in the granularity, structure, and philosophical underpinnings of upper-level ontologies. Nevertheless, several empirical studies have shown that using upper-level ontologies can improve both the quality and the efficiency of the ontology-development process [17]. In this research, BFO is investigated as the upper-level ontology. BFO has been used widely in the biological domain for integrating disparate ontologies or developing interoperable ontologies for biological applications [19]. There are several reasons that make the investigation of using BFO as an upper ontology worthwhile for many domains including the supply chain domain. Firstly, BFO has a very large user base and it is widely used in a variety of ontologies including military and intelligence. Secondly, BFO is very small, with only 35 classes, and correspondingly easy to use and easy to learn. Additionally, BFO is very well-documented and there are multiple tutorials, guidelines, and web forums for using BFO in ontological projects.

As a domain-neutral upper-level ontology, BFO adopts a *realist approach* and represents different types of entities that exist in the world and relations between them. Realism-based ontologies are formalized descriptions that are based on scientific theories about the nature of entities in reality and the relationships between them. The notion of *ontological realism* amounts to the idea that an ontology should be analogous not to a data model, but rather to a reality model [18]. This maximizes the utility and stability of the ontologies that are based on BFO because a data model can be a specific view of the reality.

By choosing to investigate BFO first as the top-level ontology does not mean that it is necessarily the best top-level ontology for representing the domain of supply chain management. In fact, what we have observed so far in the IOF is that it is difficult, if not impossible, to single out one of the aforementioned top-level ontologies that can fully meet the requirements of the IOF working groups (much less the entire industrial subdomains) without workarounds or extraneous assumptions. For example, a question often arises to whether realism precludes the descriptions of non-existing or abstract entities such as a simulation model. While according to the authors of BFO, that is not the case; it is however a subject of validation within IOF. One of the objectives of IOF WGs is to experiment with multiple foundational ontologies and evaluate their strengths and weaknesses. We expect to conduct similar supply chain reference ontology modeling with other upper-level ontologies such as DOLCE in the future. At the time of writing this paper, IOF has not committed to adopting a single top-level ontology yet; and multiple scenarios, including allowing more than one top-level ontology, merging multiple top-level ontologies, or no top-level ontology, are currently being considered.

ONTOLOGY DEVELOPMENT METHODOLOGY

To develop the supply-chain reference ontology (SCRO), we imported BFO as the single, upper ontology. BFO splits all entities into two categories: *continuants* and *occurrents*. Continuants are the entities that continue to persist through time while maintaining their identity. Occurrents are the processes, events, or happenings in which continuant entities participate. Also imported are a few other domain-independent, mid-level ontologies. Examples of mid-level ontologies include time ontology, unit of measure ontology, and geospatial ontology.

The current version of SCRO uses Common Core Ontology (CCO), which bundles these multiple, mid-level ontologies. This approach conforms with the hub-and-spoke architecture recommended in the IOF's technical principle document [20]. SCRO also imports and extends the IOF proof-of-concept (POC), top-down ontology, which contains a small set of high-level terms such as *engineered system*, *product,* and *manufacturing resource*, that are common across multiple industrial manufacturing subdomains.

Different Application Ontologies (AO) in the domain of supply chain can import SCRO and further extend it to address application-specific requirements. The validation section in this paper describes how an application ontology related to the supplier discovery and evaluation is created based on the SCRO model. This tiered architecture is shown in Figure 1.

We adopted a bottom-up approach, driven by supply chain use cases, for developing SCRO. For this purpose, a template was designed for 1) describing the problem statement, 2) identifying the expected role of an ontology in the proposed use case, 3) listing the key notions (terms) related to the use case, and also 4) providing some Competency Questions (CQ) to be used later for validation purpose. Four use cases, related to different phases of supply chain, were proposed by the members to motivate the bottom-up development process. The proposed use cases were related to supplier discovery, supply-chain configuration, bidding automation, and supply-chain traceability. Table 1 shows the details of the supplier discovery use case.

The core terms related to different use cases were aggregated to create a draft list of terms composed of about 80 entries. The informal (natural language) definitions were created based on the procedure described in Figure 2. The informal definitions are human-readable definitions of the term and they are intended to be intelligible for subject-matter experts (SME).

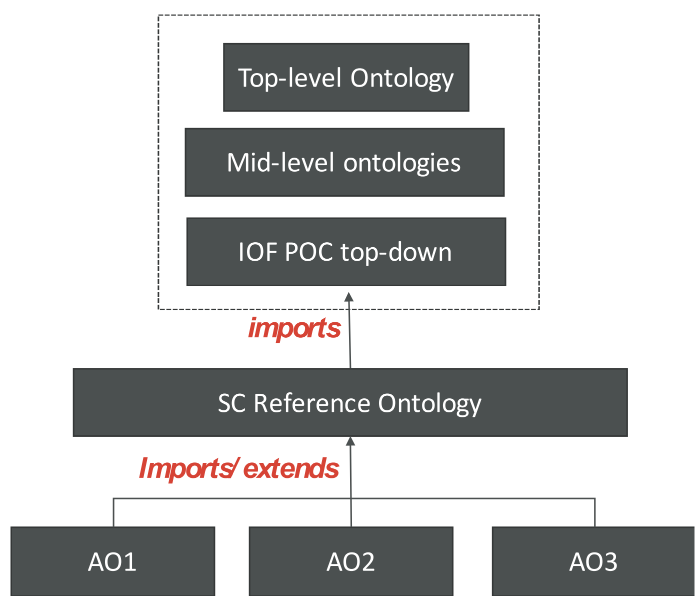

Figure 1. The position of SCRO in the stack of IOF ontologies

According to this process, the candidate definitions for each term are collected from internal sources (IOF members) and several external sources including Ontobee repository [21], ISO online browsing platform [22], relevant domain glossaries such as APICS [23]. If any of the collected definitions is deemed suitable to be used directly, it will be adopted as the informal (or Subject-Matter Expert- SME) definition as is. Otherwise, the candidate definitions will go through some linguistic and semantic pre-processing such as disambiguation, reconciling contradictions, removing unnecessary contextual contents, and removing redundancies to arrive at a more refined definition. Additional documentations were discussed in IOF including examples and corner cases.

Online sharable spreadsheets were the preliminary tool used for curating the list of terms and their definitions and also maintaining the history of their evolution. There are ongoing discussions to use better version-control systems, such as GitHub, and collaborative ontology development tools, such as Mobi [24]. Some examples of formal and informal definitions are provided in Table 2.

Table 1: Supplier Discovery use case

Problem Statement: *describe current state and future state*

Supplier discovery and search is often a manual, slow, and inefficient process. As the interaction between suppliers and customers becomes increasingly virtual and the lifespan of supply chains becomes shorter, more efficient and intelligent approaches to supplier search and evaluation are called for. One of the root causes of inefficiency in sourcing process is that manufacturing companies often publish and share their capabilities using informal and unstructured representation methods. Therefore, it is difficult to automate the sourcing or supply chain formation process.

How ontologies can help? (examples: search, data integration, decision support)

Primary utilities:

- **Decision support/inference:** Ontologies can support human experts during sourcing process by providing inference-based answers to various queries about suppliers' capabilities.

Secondary utilities:

- **Semantic Integration:** Ontologies can help with semantic integration of heterogeneous manufacturing capability models generated by dispersed actors.
- **Automation:** Ontologies can enable machine agents to actively participate in supply chain formation process by proving machine-understandable content.

Competency Questions: (include at least 5 questions)

Which factories can machine complex geometries?

What is the precision machining capability of this supply chain?

What is the minimum wall thickness that can be machined in this factory?

What is the largest diameter that can be turned in the factories owned by this company?

What is the capability of this factory with respect to surface roughness?

Does the capacity of this supply chain satisfy the demand?

Relevant terms:

• Supply chain • Supplier • Customer • Vendor • Manufacturing capability • Delivery lead time • Production capacity • Work order • Request for quote	• Sourcing • Supply chain management • Inventory • Resource utilization • Supply • Demand • Seasonal demand • Factory • Machine Cell • Capacity • Transportation mode

• Outsourcing Procurement	

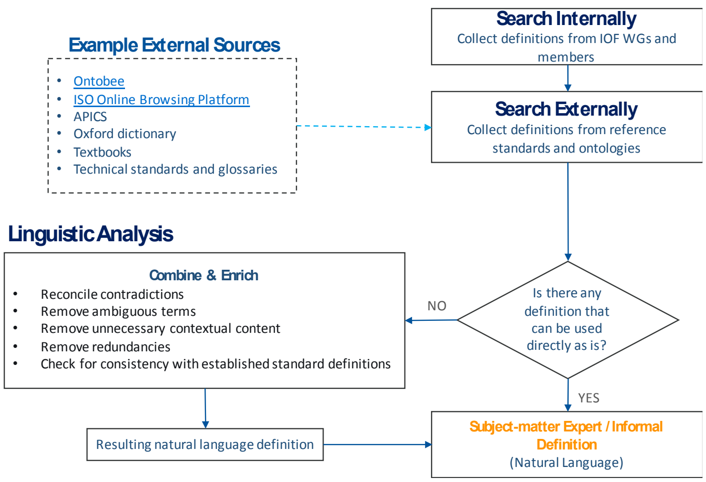

Figure 2. The overall process of creating subject-matter expert definition

Once a consensus is reached on the informal definition of a term, the ontological analysis begins by arranging the terms into a hierarchy based on their BFO or IOF classes. Then formal definitions of the terms are created. Formal definitions should use the vocabulary of the relevant upper-tier ontologies and follow the Aristotelian definition strategy, whereby each definition is of the form 'A =def. a B which Cs'. B is the parent term/class of A in the ontology, and C is the *differentia* which marks out those Bs which are As. The last step in ontological analysis is adding formal axioms (such as equivalence and sub-class axioms) to each term to create the formal expression of the term in OWL language. This process results in creation of ontological classes or universals.

SCRO Requirements:

The specific requirements for SCRO are listed below. The terms "SHOULD" and "SHALL" are to be interpreted as described in RFC2119:

- SCRO SHOULD be small and modular
- All classes in SCRO SHALL be subclasses of a class in the mid-level or top-level ontology.
- SCRO SHALL reuse the existing relationships available in mid-level and top-level ontologies to the extent possible.
- SCRO SHOULD provide the necessary generic classes that can be used for representation of different supply chain processes, roles, functions, capabilities, material entities, and information entities.
- SCRO SHOULD provide the necessary building blocks for developing supply chain application ontologies with strategic, tactical, and operation focus.
- SCRO SHOULD be sufficiently axiomatized to enable interoperability and application integration.

Table 2: Examples of formal and informal definitions of SCRO terms (CCO stands for Common Core Ontologies. The prefix indicates the source ontology for an imported term)

Term	Formal Definition	Informal Definition
Supply Chain	A CCO:GroupOfOrganizations involved in trading IOF:Products and IOF:Services and other business relationships with one another.	supply chain is a set of companies and other organizations involved in trading and other business relationships with one another.
Supplier	An IOF:Organization or IOF:Person with a IOF:SupplierRole	An organization or person who sells products or services.
Supplier Role	An IOF:SupplierRole is a BFO:Role inhering in anCCO:Agent that, if realized, is realized in some act of selling.	[no SME definition necessary since it is a construction entity and not a user-facing entity]

SUPPLY CHAIN REFERENCE ONTOLOGY

The main elements of a supply chain are the organizations that comprise the supply chain. Other elements include 1) the materials and information that flow throughout the supply chain and 2) the processes in which those material and information entities participate. This section describes how the aforementioned entities are further formalized ontologically in SCRO through definitions and axioms. These axioms are in draft statuses. First-Order Logic (FOL) notation is used for axioms shown in this paper. In FOL notation, → denotes a sub-class axiom and ≡ denotes an equivalence class axiom.

Supply Chain Material Entities

In BFO, one of the most relevant subclasses of continuants is the *material entity* class. It is a continuant that includes some portions of matter as part. Machines, physical products, raw materials, people, and organizations are examples of material entities in the domain of supply chain. Selected axioms related to organization, person, agent, supplier, customer, and manufacturing resource are illustrated in this section. Some of the presented axioms may appear to be too weak or too strong in an absolute sense. We expect that as SCRO is validated by more application ontologies, the axioms will be adjusted to meet the formalization requirement of different use cases.

organization is a subclass of bfo:object aggregate.

organization(x) → object-aggregate(x).

person subclass of bfo:object.

person(x) → object(x).

Every **agent** at time t is a **person** or **organization**.

instance-of(x,agent,t) ≡ (instance-of(x,person,t) or instance-of(x,organization,t)).

supplier and **customer** are subclasses of agent

supplier(x) → agent(x).

customer(x) → agent(x).

The axioms related to *role classes* usually contain the notion of time (t) because entities can bear different roles in different time intervals.

A **supplier** is an agent who bears a supplier role.

Instance-of-supplier(x, supplier,t) ≡ ∃y(supplier-role(y) & has-role(x,y,t)).

A **manufacturing supply chain** is a group of suppliers connected by supply chain links to manufacture a certain product.

A **supply chain link** (a sub-property of *participates-in*) is a relationship between two suppliers (s1 and s2) if s participates in the process of supplying material and information to s2.
The axiom that can be used for querying if a supplier (s1) is a member of a supply chain (sc1) is as follows:

supplier (s_1) *& supply chain* (sc_1) *& (∃* s_2*(supplier(*s_2*) & has-supply-chain-link (*s_1*,* s_2*) & member-of (*s_2*,* sc_1*)) → member-of (*s_1*,* sc_1*)*

manufacturing resource is a bfo:continuant that bears a manufacturing resource role.

instance-of (x,manufacturing resource,t) ≡ ∃y(manufacturing resource role(y) & has-role(x,y,t)).

A **piece of manufacturing equipment** is a piece of equipment that bears a function and any process which realizes that function is a manufacturing process.

piece of manufacturing equipment(x) ≡ piece of equipment(x) & ∃f(has-function(x,f) & ∀p(process(p) & realizes(p,f))→ manufacturing process(p)).

Supply Chain Roles

The *role* class in BFO provides a versatile template for creating different *defined classes* when an entity is in some special natural, social, or institutional set of circumstances. Role is a *realizable entity* in BFO, meaning that it is only manifested or realized in certain conditions and certain times. Examples of supply chain roles include 1) transportation role that is the role of an organization to serve as a provider of transportation service in a supply chain or 2) product role that is the role of an artifact to be sold or exchanged in a supply chain. Role may also be viewed as a work around to maintain a single inheritance hierarchy. In other words, an entity is born or created or designed to be what it is (the class in which is asserted to a member) but can engage in many roles. The **supply chain member role** is used as the top class for different types of supply chain roles such as material provider role or test service provider role. Figure 3 shows some of the sub-types of supply-chain-member role. The formal definitions of important supply chain roles, together with their axioms, are provided below.

supplier role subclass of role.

supplier-role(x) → role(x).

A **supplier role** is a role inhering in an agent that, if realized, is realized in some act of selling.

supplier-role(x) ≡ ∃y(agent(y) & has-role(y,x) & ∃p(process(p) & realizes(y,p) → act-of-selling(p))).

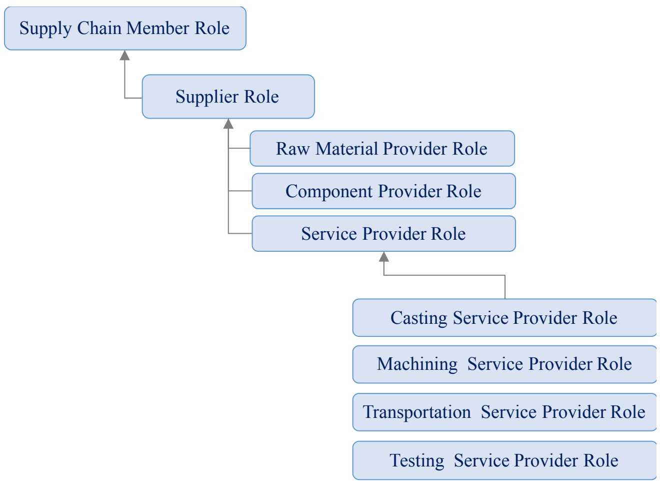

Figure 3. Different sub-classes of supply chain member role class

Note that the supplier role is only related to the *act of selling* (a product or a service). There is no mention of the particular product that will be produced as a result of the participation of the supplier. There are three different sub-types of supplier role in SCRO, namely, raw material provider role, component provider role, and service provider role.

A **service provider role** is a role inhering in an Agent that is realized in some act of service provision. A **machining service provider role** is a role inhering in an agent that, if realized, is realized in some act providing a machining service.

Machining-service-provider- role(x) $\equiv$ *∃y(agent(y) & has-role(y,x) & ∀p(process(p) & realizes(p,x) → act-of-providing-machining-service(p)).*

Accordingly, a **machining supplier** is a supplier that bears a machining service provider role.

machining supplier(x) $\equiv$ *supplier (x) &(∃y(machining service provider role(y) & has-role(x,y)).*

The **customer role** is the role inhering in an agent in a basic economic transaction from the point when a purchasing act has been initiated through to completion.

instance-of(x,customer-role,t) $\equiv$ *∃y,z(agent(y,t) & has-role(y,x,t) & ∃w(instance-of(w,act-of-purchasing,t) & agent-in(y,w,t))).*

Figure 4 shows the function, capability, and the role related to a *Machining Supplier*. In this figure, the green boxes are BFO classes, while the blue boxes denote SCRO classes. The white boxes represent the classes imported from mid-level ontologies such as IOF top-down ontology or CCO. Figure 5 shows the class structure of some of the SCRO continuants.

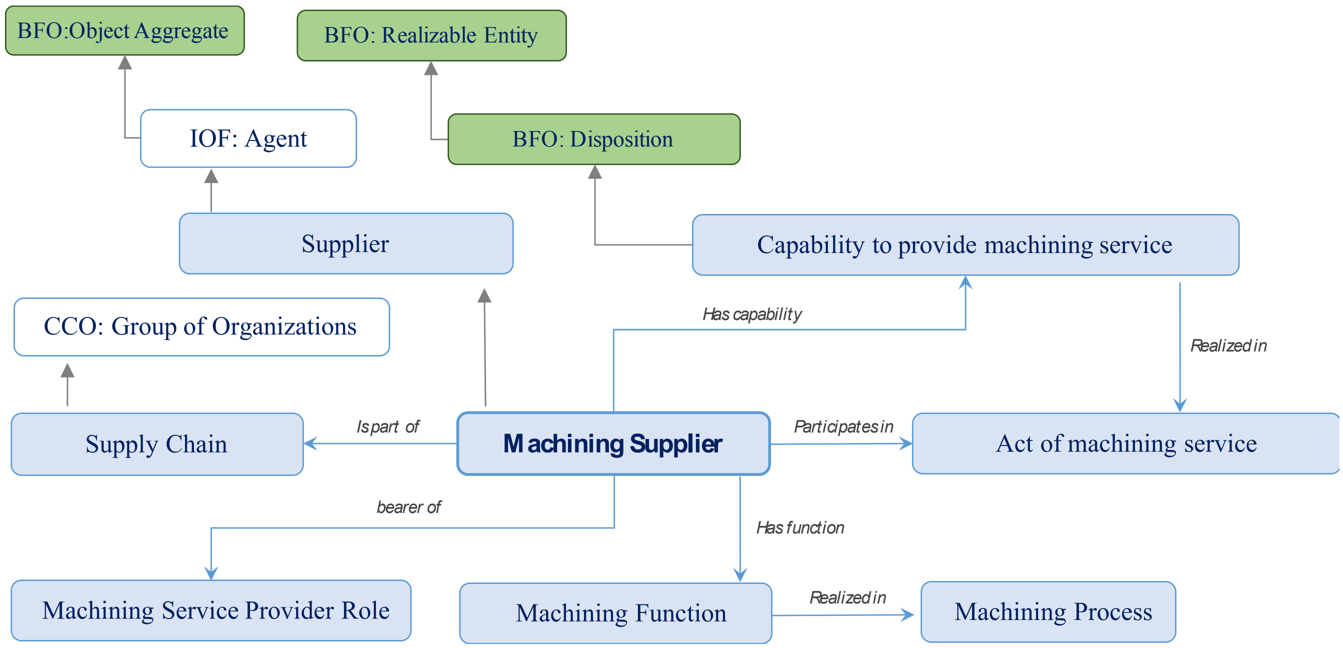

Figure 4. The class diagram for the Machining Supplier class

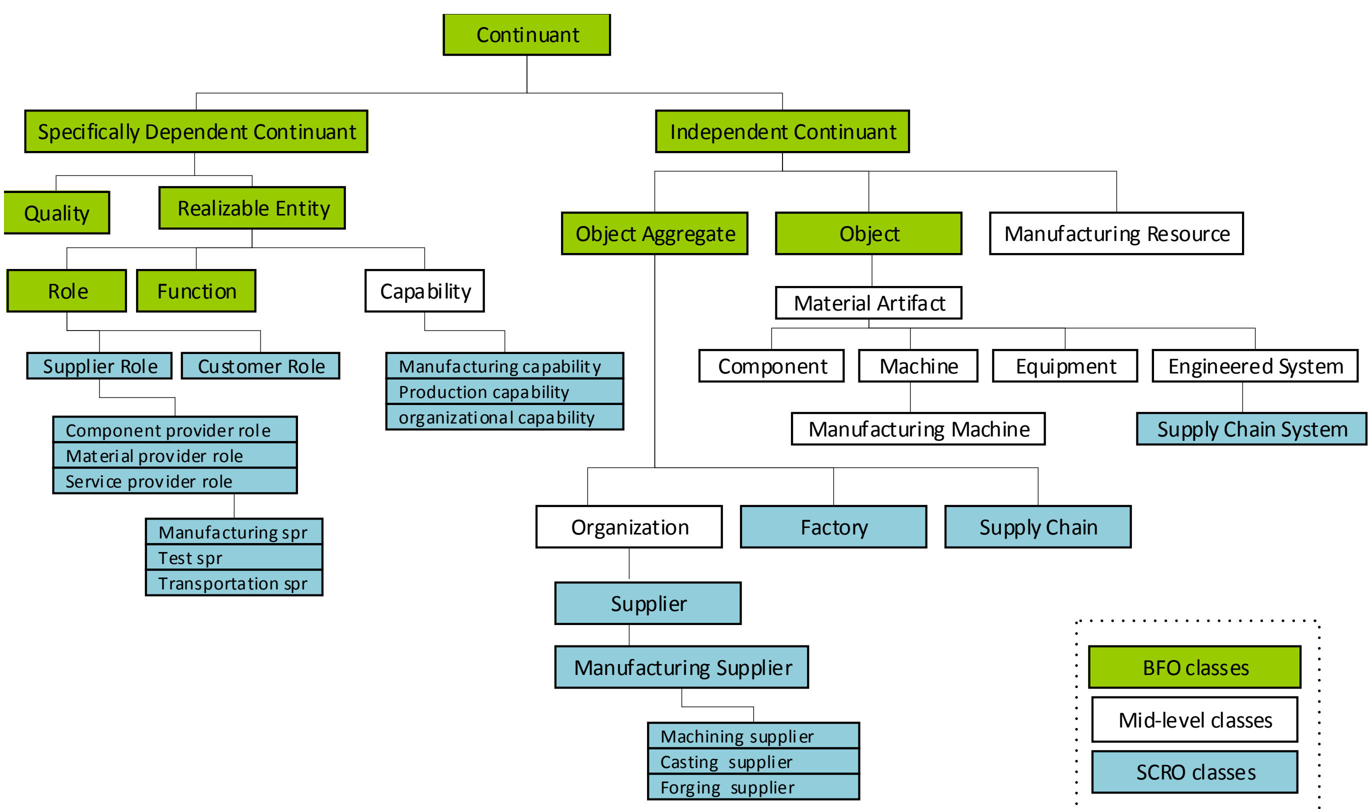

Figure 5. The continuant side of SCRO (partial view)

Network Representation of Supply Chains

Supply chains can be perceived as a network in which the nodes represent the partnering organizations and the edges denote the flow of material and information between the organizations (Figure 6).

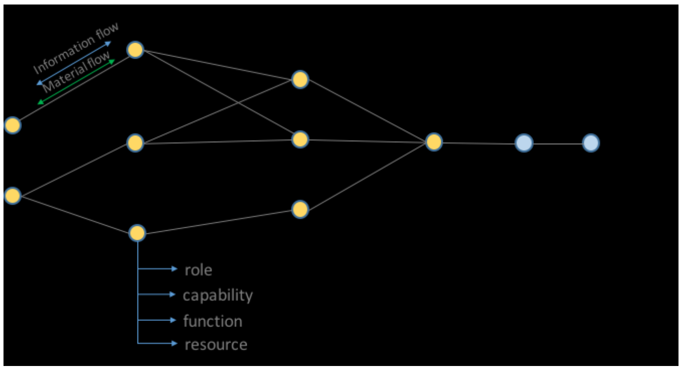

Figure 6. Network representation of supply chain

The organizations within a supply chain have different roles, functions, capabilities, and resources. SCRO, as a reference ontology, provides the necessary building blocks for modeling supply chains as a network. That is, the supply chain link, that was defined in the previous section, represents an edge in the network. A particular supply chain then can be defined as a group of organizations that participate in production of a specific product. The material entities, such as different types of raw materials, products, and semi-finished assemblies, that flow through the network have their own ontological representation.

As shown in

Figure 7, the flow of material between two suppliers is modeled as a participation of a material entity in an *Act of Shipment*. Act of Shipment (an Occurrent) is a sub-class of Act of Location Change in Common Core Ontology (CCO). An Act of Location Change is defined as *An Act of Motion*, in which the location of an Object is changed by some Agents. Supplier 1 (S1) participates in the act of shipment as the sender of the material entities and Supplier 2 (S2) participates in this act as the receiver of those material entities sent by S1.

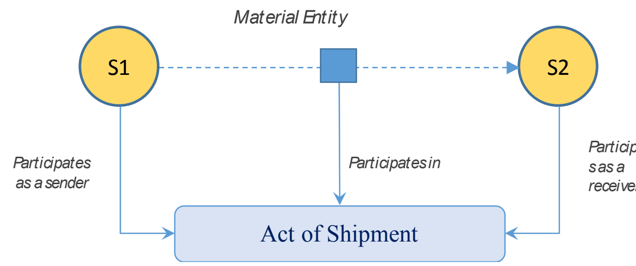

Figure 7. Flow of materials between two nodes of the supply chain: supplier 1 (S1) and supplier 2 (S2).

Using a similar approach, information communication can be modeled as an *Act of Communication* in with the participation of a sender (S1) and a recipient (S2) participate. The Information Entity itself is one the participants in the act of communication.

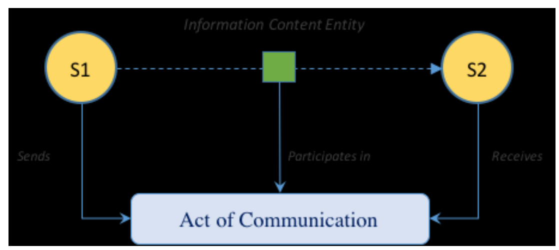

Figure 8. Flow of information entities between two nodes of the supply chain: supplier 1 (S1) and supplier 2 (S2).

The definitions for *sends* (information) and *receives* (information) properties are adopted from Common Core Ontology:

- **sends** (*inverse of* **has sender**) is a relationship between an Agent *a1* and an Act Of Communication *c1* and such that *a1 sends c1* if and only if *a1* is the initiator and encoder of the Information Content Entity participating in *c1*.
- **Receives** (*inverse of* **has recipient)** is a relationship between an Agent *a1* and an Act Of Communication *c1* such that *a1 receives c1* if and only if *a1* is the **recipient and decoder** of the Information Content Entity participating in c1.

When a network model of the supply chain is available, interesting information can be derived through reasoning and querying the network in order to answer important business questions (competency questions) such as:

- What are the outputs of this supplier in this supply network?
- What types of information does this supplier receive?

- What is the most connected supplier in this supply network?
- What are the first-tier/second-tier suppliers in this network?
- Which supplier provides raw materials in this network?
- What is the path followed by a certain component in this network?

SUPPLIER DISCOVERY USE CASE

To validate the proposed reference ontology, we focused on a supplier sourcing use case. Various software agents collaborate and interact with the end goal of forming a supply chain for a given production order. Consequently, interoperability is an essential feature of this use case since the agents operate independently as autonomous entities. For the purpose of this use case, SCRO was extended to create an *application ontology* tailored based on the needs of an agent-based sourcing scenario. The participating machine agents in this use case provide different types of web services as described below:

- **Capability Advertisement Service (CAS):** Each manufacturing company is represented by a CAS agent which responds to queries regarding the manufacturing capabilities of the company. The capabilities are modeled using the extensions of the "capability" module of SCRO. In this use case, it is assumed that the capability data is coded as ontology instances. Another plausible scenario is to keep the capability data in data structures that are tagged using ontological entities.
- **Supplier Evaluation Service (SES):** The Supplier Evaluation Service evaluates suppliers with respect to their abilities in fulfilling the requirements of specific production orders. SES receive requests from the Supply Chain Configuration agents. Different SES agents may use different methods and algorithms for supplier evaluation.
- **Supply Chain Configuration Service (SCCS):** SCCS agents are in charge of building a supply chain that can complete a production order and manufacture the requested components or assemblies in the requested volumes and delivery times per specifications. They interact and communicate with SES agents to identify qualified suppliers that can participate in the desired supply chain.
- **Supply Chain Evaluation Service (SCES):** SCES agent evaluates the performance of a supply chain using key performance measures such as reliability, responsiveness, and cost.

In this use case, an ontology can serve two purposes: 1) to enable interoperability among heterogenous agents and 2) to enable capability analysis and inference based on the explicitly stated capabilities. Figure 8 shows how the subclasses of the *capability* class are extended for this use case. SCRO contains three types of capability classes, namely, manufacturing capability, production capability, and organizational capability. The manufacturing capability class was extended to represent the classes needed for asserting the capabilities related to attributes of manufactured artifacts. For example, *part material capability* is related to the capabilities of a certain organization to process different types of materials. Similarly, *part complexity capability* is the class related to range of geometric complexities that can be accommodated by a manufacturer.

Capability Representation

The CAS agent can represent and advertise the capabilities of a manufacturing company both directly and indirectly. Direct capability representation entails providing values for different measures of manufacturing capability such as complexity, material, and precision capability measures. Figure 10 shows the template used for expressing the value of surface roughness capability. According to this template, *Part Surface Roughness Capability* is a capability that is measured as a *length measurement datum* which is a type of *Measurement Datum*, a class imported from the Information Artifact Ontology (IAO).

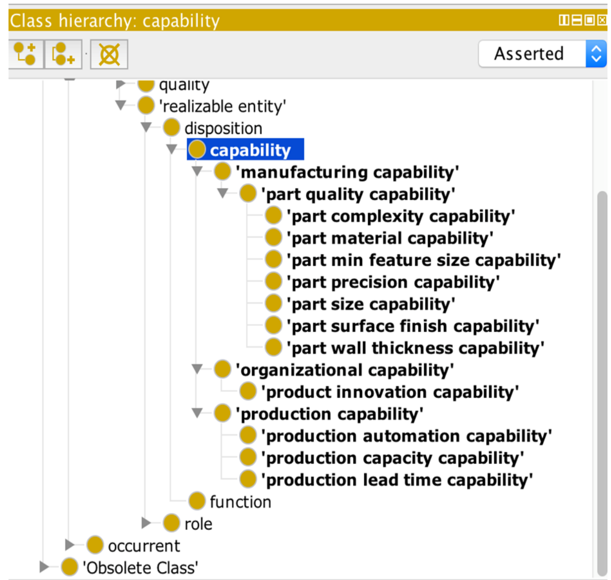

Figure 9: The extended capability class

Indirect representation of capabilities involves two steps. First, we describe the manufacturing resources owned by the company. Second, we allow the supplier-evaluation agents to interpret the capabilities based on the available resources. The capability inference methods described in the next section are based on indirect capability descriptions. In doing so, we assume that the company instantiates the *factory* class, which is considered to be an *object aggregate* in SCRO. The manufacturing resources are linked to the factory class using *has part* relationship.

In many occasions, capability-related queries can be formulated as SPARQL queries. For example, the query shown in Figure 11 returns all factories that can provide vertical-milling services for parts that are longer than 10 inches in length. However, more sophisticated reasoning might be needed when, for example, the supplier-evaluation agent is interested in learning about different ranges of part complexities that can be supported by a given supplier. In this example query, MSD [26] is imported as an external ontology to enhance the expressiveness of the application ontology which is developed through extending SCRO.

Capability Inference

One of the functions of the Supplier Evaluation Agent is capability inference. Using the extended capability module of SCOR, one can create a formal representation of a factory. From the capabilities explicitly represented in the factory model, new capabilities can be inferred automatically using the ontological reasoning services. Four categories of capability are discussed in this section: 1) surface roughness capability, 2) process capability, 3) material capability, and 4) production capability. It should be noted that such inferences at best provide an approximation of the latent capabilities of a supplier.

Figure 10: Surface roughness capability measurement template

Active Ontology × | Entities × | Individuals by class × | DL Query × | SPARQL Query ×

SPARQL query:

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX obo: <http://purl.obolibrary.org/obo/>
PREFIX msdl: <http://infoneer.txstate.edu/ontology/>

SELECT ?factory_instance
WHERE
{
   ?factory_instance a msdl:factory .
   ?vertical_milling_capability_instance a msdl:"vertical milling capability" .
   ?factory_instance msdl:MSDL_0000768 ?vertical_milling_capability_instance .
   ?length_measurement_datum_instance a obo:IAO_0000408 .
   ?length_measurement_datum_instance obo:IAO_0000004 ?length_value .
   ?part_length_capability_instance a msdl:MSDL_0000674 . #
   ?part_length_capability_instance msdl:MSDL_0000924 ?length_measurement_datum_instance .
   ?production_machine_instance a/rdfs:subClassOf* msdl:MSDL_0000033 . # "
   ?production_machine_instance msdl:MSDL_0000936 ?part_length_capability_instance .
   ?factory_instance msdl:MSDL_0000129 ?production_machine_instance .
    FILTER(?length_value > 10)  }
```

Figure 11: A SPARQL query returning all factories with vertical milling capabilities for parts longer than 10 inches

Surface finish capability:

A machine tool can create certain qualities such as tolerance, surface roughness, or minimum feature size on a part. The range of these qualities define the capability of the machine tool. The collective capability of the factory is calculated by aggregating the capabilities of individual machines in the factory. Figure 12 shows the procedure for calculating the surface-finish capability that can be provided by a factory operated by a given company. According to this procedure, each machine's surface-finish. capability value, which is already stored as instance information, is retrieved first. If the retrieved value is null, then the immediate superclass of the machine tool is queried instead and the surface -finish capability value is retrieved. The reason behind this approach is that the parent type of machine class can provide a reasonable approximation of the capabilities of the children types of machines. If none of the higher-level individuals can provide a value for surface- finish capability, then a similar, generic machine from the same machine vendor is used to provide some approximation about the capability of the machine. The generic machine from a given vendor is the average machine with respect to capabilities based on the vendor's product portfolio.

FOR i=1 to num of machines in the factory f [instance of SCRO: factory]
m_i [instance of SCRO: machine tool]
Retrieve **part surface finish capability** value $m_i\_ sf_{cap}$
IF m_i- sf_{cap} = Null
    Find an instance of the superclass of m_i => sm_i
Retrieve **part surface finish capability** value $sm_i\_ sf_{cap}$
    IF sm_i _ sf_{cap} = null THEN
    Find the instance of a **Generic Machine** from the same vendor => gm_i
        Retrieve **part surface finish capability** value gm_i _ sf_{cap}
            IF gm_i- sf_{cap} = Null THEN Let m_i- sf_{cap} = null
                ELSE Let m_i- sf_{cap} = gm_i- sf_{cap} AND go to the next machine
                End IF
        ELSE Let m_i- sf_{cap} = sm_i _ sf_{cap} AND go to the next machine
        End IF
End IF
Factory - sf_{cap} =Min (m_i- sf_{cap}) for all values of i

Figure 12: The procedure of calculating the surface finish capability of a factory

This procedure is based on the simplifying assumption that surface-finish capability is a standalone capability. However, more realistically, surface-finish capability is related to other types of capability such as surface-area capability or material capability.

Material Capability:

Material capability is also inferred based on the *is-a* relationship between different instances of materials. The factory model contains a list of materials that can be processed by the factory. The ontological reasoner can identify all super-classes of the explicitly stated material types. The instances of the identified upper- level classes are then added to list of materials that can be processed at the factory as inferred materials.

The logic behind this approach is that if a particular vertical mill, for example, can machine a special grade of aluminum then it can also machine more generic grades of aluminum as well. Material capability, in most real-life scenarios, is evaluated in relation with other capabilities. For example, the grade of material may impact the achievable tolerances and surface finishes on a given machine tool. These dependencies between capabilities can be encoded in the ontology through defining semantic rule which is outside the scope of this paper.

Process Capability:

A manufacturing process is the realization of an explicit manufacturing function intended for a piece of manufacturing equipment. When instances of manufacturing equipment are added to the factory, the manufacturing functions associated with the equipment are added to the list of available functions in the factory. The functions added directly through the equipment are considered to be explicit functions. The ontological reasoner identifies all sub-classes of the explicit functions as the inferred functions. For example, if a machine in a factory has a 'turning' function, then all instances of sub-classes of turning (including boring, facing, grooving, threading) are added to the list of 'inferred' processes (functions) for that factory.

Production Capability:

Production capability of a manufacturing facility is related to factors such as the number of equipment and the variety of the products that can be produced. Simplistically, the capacity of the factory directly depends on the bottleneck resource in the factory. There are some other indirect factors, such as the availability of the preventive maintenance system that can alter the capacity.

The capacity-capability class can be measured as an ordinal measurement datum with low, medium, high values. The variety capability, also measured as an ordinal measurement datum, depends on the available types and variety of manufacturing processes. Availability of more manufacturing functions can imply a higher variety of manufacturable parts. Therefore, the reasoner should consider both the explicit and the inferred processes when calculating the variety capability of a factory.

Figure 13: Proof-of-concept tool for capability inference

Based on the described procedures, a proof-of-concept tool was developed for 1) analyzing different capabilities of manufacturing suppliers and 2) inferring new capabilities based on the explicit capabilities. Figure 13 shows one of the user interfaces of the developed tool related to process capability analysis.

CLOSING REMARKS

In this paper, we presented the initial version of a reference ontology (SCRO) that can be used for creating more specialized application ontologies for the supply-chain domain. SCRO was developed based on the methodologies and specifications recommended by IOF. BFO was used as the top-level ontology. One of the objectives of the current experiment was to evaluate BFO as an upper ontology for the supply-chain domain. Different criteria such as ontological completeness, logical consistency, and accuracy in capturing the domain were used to perform that evaluation.

The results of the evaluation indicated that the breadth of coverage of BFO is adequate to represent commonly used entities in the supply chain domain. Also, the underlying axioms of the top-level ontology provided the needed logical consistency for the investigated use cases. However, further testing is needed to evaluate the adequacy of BFO for more complicated use cases, in which interoperability and applications integration are the main concerns.

One of the issues related to BFO is its inaptitude to represent the abstract notions that is rooted in its realist approach. Examples of abstract notions in supply-chain domain include the various supply chain-models that do not exist but need to be represented; e.g., for simulation and analysis. Workarounds are often needed to address abstract notions in BFO. In the future, addtional application ontologies will be created based on SCRO to evaluate its adequacy for different use cases that require interoperability and inference services.

DISCLAIMER AND ACKNOWLEDGEMENT

Certain commercial systems and applications identified in this paper are not intended to imply recommendation or endorsement by the National Institute of Standards and Technologies, nor is it intended to imply that they are necessarily the best available for the purpose. We acknowledge the input we received from the IOF community.

REFERENCES

[1] Kim, H. M., and Laskowski, M., 2018, "Toward an ontology-driven blockchain design for supply-chain provenance," Intelligent Systems in Accounting Finance & Management, 25(1), pp. 18-27.

[2] Tian, K. S. a. Y., "Supply Chains Integration: Architecture and Enabling Technologies," Journal of Computer Information Systems(- 3), pp. - 67.

[3] Chapurlat, V., and Daclin, N., 2012, "System interoperability: definition and proposition of interface model in MBSE Context," IFAC Proceedings Volumes, 45(6), pp. 1523-1528.

[4] Rayyaan, R., Wang, Y., and Kennon, R., 2014, "Ontology-based interoperability solutions for textile supply chain," Advances in Manufacturing, 2(2), pp. 97-105.

[5] Arp, R., Smith, B., and Spear, A. D., 2015, Building Ontologies with Basic Formal Ontology, The MIT Press.

[6] Guarino, N., Oberle, D., and Staab, S., 2009, "What Is an Ontology?," pp. 1-17.

[7] Fox, M. S., Barbuceanu, M., and Gruninger, M., 1996, "An organisation ontology for enterprise modeling: Preliminary concepts for linking structure and behaviour," Computers in Industry, 29(1), pp. 123-134.

[8] Soares, A. L., Azevedo, A. L., and de Sousa, J. P., 2000, "Distributed planning and control systems for the virtual enterprise: Organizational requirements and development life-cycle," Journal of Intelligent Manufacturing, 11(3), pp. 253-270.

[9] Scheuermann, A., and Leukel, J., 2014, "Supply chain management ontology from an ontology engineering perspective," Computers in Industry, 65(6), pp. 913-923.

[10] Geerts, G. L., and O'Leary, D. E., 2014, "A supply chain of things: The EAGLET ontology for highly visible supply chains," Decision Support Systems, 63, pp. 3-22.

[11] Grubic, T., and Fan, I.-S., 2010, "- Supply chain ontology: Review, analysis and synthesis," Computers in Industry, 61(-8), pp. 776-786.

[12] 2019, "IOF Charter," https://www.industrialontologies.org/iof-charter/.

[13] Kulvatunyou, B., Wallace, E., Kiritsis, D., Smith, B., and Will, C., 2018, "The Industrial Ontologies Foundry Proof-of-Concept Project," IFIP WG 5.7 International Conference, APMS 2018, Springer, Seoul, South Korea.

[14] Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, R., Schneider, L., and Partner Istc-cnr, L., 2002, WonderWeb Deliverable D17. The WonderWeb Library of Foundational Ontologies and the DOLCE ontology.

[15] Gruninger, M., and Menzel, C., 2003, "The Process Specification Language (PSL) Theory and Applications," AI Magazine, 3(24).

[16] Niles, I., and Pease, A., 2001, "Towards a standard upper ontology," Proceedings of the international conference on Formal Ontology in Information Systems - Volume 2001, ACM, Ogunquit, Maine, USA, pp. 2-9.

[17] Keet, C. M., "The Use of Foundational Ontologies in Ontology Development: An Empirical Assessment," Proc. The Semantic Web: Research and Applications, G. Antoniou, M. Grobelnik, E. Simperl, B. Parsia, D. Plexousakis, P. De Leenheer, and J. Pan, eds., Springer Berlin Heidelberg, pp. 321-335.

[18] Smith, B., and Ceusters, W., 2010, "Ontological realism: A methodology for coordinated evolution of scientific ontologies," Applied Ontology, 5(3-4), pp. 139-188.

[19] Hoehndorf, R., Schofield, P. N., and Gkoutos, G. V., 2015, "The role of ontologies in biological and biomedical research: a functional perspective," Briefings in Bioinformatics, 16(6), pp. 1069-1080.

[20] 2019, "IOF Technical Principles Document," https://www.industrialontologies.org/iof-technical-principles-document/.

[21] Ong, E., Xiang, Z., Zhao, B., Liu, Y., Lin, Y., Zheng, J., Mungall, C., Courtot, M., Ruttenberg, A., and He, Y., 2017, "Ontobee: A linked ontology data server to support ontology term dereferencing, linkage, query and integration," Nucleic acids research, 45(D1), pp. D347-D352.

[22] 2019, "ISO Online Browsing Portal (OBP)," https://www.iso.org/obp/ui/.

[23] Jr., J. H. B., 2013, "APICS Dictionary," APICS, Chicago, IL.

[24] 2019, "Mobi," https://mobi.inovexcorp.com.

[25] Smith, B., and Ceusters, W., 2015, Aboutness: Towards Foundations for the Information Artifact Ontology.

[26] Ameri, F., and Dutta, D., "An upper ontology for manufacturing service description," Proc. 2006 ASME International Design Engineering Technical Conferences and Computers and Information In Engineering Conference, DETC2006, September 10, 2006 - September 13, 2006, American Society of Mechanical Engineers.

