Advanced Engineering Informatics 38 (2018) 129–141

Contents lists available at ScienceDirect

Advanced Engineering Informatics

journal homepage: www.elsevier.com/locate/aei

Full length article

A foundational ontology for the modelling of manufacturing systems

Viktor Zaletelj[a], Rok Vrabič[b], Elvis Hozdić[b], Peter Butala[b,*]

[a] L-TEK Elektronika d.o.o., Obrtna cesta 18, SI-8310 Šentjernej, Slovenia
[b] University of Ljubljana, Department of Control and Manufacturing Systems, Aškerčeva 6, SI-1000 Ljubljana, Slovenia

ARTICLE INFO

Keywords:
Manufacturing system
Modelling
Collaborative environment
Ontology

ABSTRACT

Models of distributed manufacturing systems cannot be consistent without a formal ontology. In this paper, the ontology formulation and maintenance are addressed in the scope of a collaborative modelling environment – in which concurrency, consistency, and model life cycle management should be supported. Thus, an extensible foundational ontology for manufacturing – system modelling is proposed in which the formal definitions of the modelling environment itself enable the definition of the manufacturing system's elements. The presented approach ensures the consistency of ever-changing models. The ontology is integrated into a modelling framework through the concept of description layers that assist in the management of the model description's complexity. The feasibility of the approaches is illustrated in an industrial case study that models of a manufacturing system for material processing.

1. Introduction

In the global competitive environment, manufacturing companies are forming networks to increase their agility, adaptability, innovation potential and competitiveness. The next generation of manufacturing systems, which will emerge from these networks, will be distributed and temporary. They will only be formed to satisfy a specific need, such as making a new product, and will disband afterwards. As such, they should be seen as products themselves [1,2]. A proper methodological support for their modelling, design, and control is needed.

Several issues need to be addressed in this scope. Since the manufacturing system is distributed, its modelling should be independent in terms of both location- and time. Furthermore, since the model changes dynamically over time, its consistency should be maintained during the whole life cycle, including the physical realization and operations of the modelled system.

In a distributed environment, several experts from different domains work on a model; therefore, the modelling environment should be independent of modelling methodologies. To achieve this, a common ontology does not seem to be sufficient, nor is it the most efficient way. Only a formulation of the basic ontological concepts along with the rules for their extension provides an environment where models will be able to evolve, preserving their consistency over time.

Consider a case in which several experts from different parts of the world are involved in building a model of a distributed manufacturing system. Because of their different backgrounds, their understanding of the same terms might differ. They may be using different natural or computer languages, and even if they use English, they might not agree on the meaning of some terms: what some consider to be "manufacturing", others might consider to be "production" (and vice versa). They might also want to use different tools to do the modelling. A common way of solving this issue is to agree on a common set of tools and languages, and to form a common ontology. This, however, represents an effort, and does not necessarily ensure the consistency of the models.

The paper explores an alternative approach where formal definitions of the modelling environment itself enable a definition of the elements of a manufacturing system's model. A foundational ontology is provided, on which domain-specific modelling ontologies can be formulated. The integration of the ontology into a modelling environment is formalized through the concept of description layers. The model consistency over time is ensured through the use of the basic definitions for model elements. Thus, continuous adaptation of the manufacturing system's model to changes and disturbances in the environment is enabled.

2. Literature review

In information science, ontology is defined as a method of representing items of knowledge (ideas, things, facts, etc.) in such a way that determines the relationships and classifications of the concepts within a specified domain of knowledge [3]. It provides unambiguous

* Corresponding author.
E-mail address: peter.butala@fs.uni-lj.si (P. Butala).
URL: http://www.lakos.fs.uni-lj.si (P. Butala).

https://doi.org/10.1016/j.aei.2018.06.009
Received 24 May 2016; Received in revised form 21 June 2017; Accepted 21 June 2018

definitions of the terms that can be used by software systems, as well as software users. According to Gruber [4], ontology is an explicit specification of a conceptualization. An ontology, as defined in [5], is a formal specification of a shared conceptualization of a domain of interest to a group of users.

Many views exist on the purpose of ontologies. While they aim to solve problems of poor communication, interoperability, information sharing, and information reuse, an important view is that ontologies are able to integrate models of different domains into a coherent framework [6]. This is especially important in a distributed environment where nodes might have different viewpoints on the same system as a consequence of different contexts.

Ontologies have an important role in enabling the interoperability between heterogeneous sources, as well as support in the creation of the mutual responses in the use of knowledge between different domains [7,8]. However, in the same way as an ontology provides a shared understanding of a domain, a shared understanding of the ontology itself has to be developed.

In the manufacturing domain, ontologies have received attention from several different perspectives. Today there exist approaches to the description of manufacturing ontology from different conceptual views, including product configuration modelling, product data exchange, product lifecycle management, modelling of flexible and reconfigurable manufacturing systems, machining process modelling, etc.

The product lifecycle management is addressed in [9] and [10]. In [9], a concept of core product model (CPM) is proposed for supporting product behaviour modelling during its lifecycle. CPM is a generic, abstract model with generic semantics and it is defined as a UML class diagram. In [10], a multi-level product information modelling framework is proposed, which enables stakeholders to define product models and relate them to physical or simulated instances.

The importance of ontologies and semantic web for manufacturing companies which operate in dynamic environments is outlined in [11]. Besides, the Ontology Web Language (OWL) and its limitations are discussed.

The report [12] points out the necessary integration of the ontological approaches in terms of products, processes and resources. This is especially important for managing information in loose collaboration networks due to the lack of semantic precision, ambiguity risks and usage of pre-determined context of communication [12].

Bock et al. [13] integrate ontological and model-based techniques for collaborative design. An ontology is used to capture alternative designs and incremental refinements that meet requirements and earlier design commitments.

In the domain of enterprise and organizational modelling, the Enterprise Ontology [14] and TOVE [15] provide basic definitions along with the ability to infer over information using the first order logic. The ADACOR ontology [16] uses the DOLCE foundational ontology [17] to address the definitions on the shop floor level [18]. It is based on the Holonic Manufacturing Systems paradigm, and it is built upon a set of autonomous and cooperative holons, each one being a representation of a manufacturing component or a logic entity.

Product-design ontologies to support system interoperability within manufacturing enterprises and supply chains are proposed in [19]. A product-centric supply chain ontology framework for facilitating the interoperation between all enterprises' applications involved in extended supply chain interactions is developed. Other examples of supply-chain ontologies are shown in [20], where a rigorous and systematic attempt to identify and synthesise the research in the domain of supply chain ontology from a *'philosophy of science'* perspective is given.

The product and processes development resources capability (PPDRC) ontological model is proposed in [21], which represents resource capabilities in the product- and processes- development process. The model integrates concepts belonging to different ontological theories and incorporates the concepts representing the social and agentive character of the resources, which are crucial for collaborative processes.

The process-specification language (PSL) [22] defines a neutral representation for manufacturing processes that supports automatic reasoning. It is written in Common Logic, a framework for a family of logic languages, based on the first-order logic.

Another domain-specific example is ISO 10303 (STEP), which provides a common ontology for product- manufacturing information exchange between CAx systems [23]. Some formal ontology research has considered the interoperability between design and production and provided some useful input to understanding the requirements.

For cyber-physical systems, Petnga and Austin [24] develop a new ontological-based knowledge and reasoning framework for decision support enabling the development of determinate, provable and executable models of cyber-physical systems.

Garetti and Fumagalli [25] propose a definition of a manufacturing system's structural entities on three layers related to the physical, technological and control aspects. On this basis, a control architecture integration of the ontology with web-service technologies, which enable easy configuration and reconfiguration of the manufacturing control systems, is developed [26].

In addition, efforts to use the Ontology Web Language for authoring ontologies in the manufacturing domain have been made. Examples include MASON [27], a top-level ontology for the manufacturing domain based on three main concepts, i.e., entities, operations and resources, and the AsD ontology [28] for assembly design.

The aforementioned approaches use conventional notations for ontology representation such as database diagrams, UML diagrams, and first-order logic. While these approaches are suitable for certain domains, they lack the means for describing the relations between the domains, which is vital in a distributed environment. Although the topics of ontology mapping and ontology alignment have received a lot of attention [29–32], almost no approaches discuss these topics from a temporal perspective [33].

Hepp [33] dubs ontology evolution in time 'conceptual dynamics' and identifies it as an obstacle for ontology building and adoption. Most domains exhibit at least some conceptual dynamics, i.e., the model of the domain changes in time. Ontologies have to be able to adapt to the changes in real time if the models are to be consistent with the modelled system. Not only must the ontology include concept changes, but also the changes of relations between the existing concepts.

In the example of the UML language, consistency is a well-defined term [34]. Nevertheless, only a handful of studies focus on the temporal aspects of consistency. Ndiaye et al. [35], for example, propose an extension to the language in order to manage the temporal aspects of model consistency.

Temporal aspects of ontology management, termed 'ontology evolution' have, however, received recent attention from several authors. Sassi et al. [36] propose an anticipatory approach towards ontology consistency to ensure the traceability of changes and the conformity of the ontology in relation to its original objectives. They propose that inconsistencies are detected using a set of coherence rules that the ontology must conform to. Corrective operations must be taken immediately when an inconsistency is detected. In this way the analysis of ontology consistency is made at every change. A similar approach is formally described in [37], where three types of ontology consistency are defined: structural consistency, logical consistency and user-defined consistency.

Maedtche et al. [38] consider the evolution of multiple ontologies on the Semantic Web and define a formal approach towards maintaining their consistency. They also define a six-step ontology-evolution process [39] consisting of capturing, representation, semantics of change, implementation, propagation and validation.

Jin et al. [40] analyse ontology evolution through a ripple-effect analysis. Ma et al. [41] propose a construct called 'Prioritized Knowledge Base' to describe the evolution of ontologies with conflicting information. In the case of a conflict, rule preferences are taken into account. In the case that rule preferences with respect to another rule are