International Journal of Production Research, *51*(22), 6553-6572, 2013.
doi:10.1080/00207543.2013.801570

Towards a formal manufacturing reference ontology

Zahid Usman, R.I.M. Young, Nitishal Chungoora, Claire Palmer, Keith Case and J.A. Harding

Wolfson School of Mechanical & Manufacturing Engineering, Loughborough University, Loughborough, UK

Due to the advancement in the application of Information and Communication Technology (ICT), manufacturing industry and its many domains employ a wide range of different ICT tools. To be competitive, industries need to communicate effectively within and across their many system domains. This communication is hindered by the diversity in the semantics of concepts and information structures of these different domain systems. Whilst international standards provide an effective route to information sharing within narrowly specified domains, they are themselves not interoperable across the wide range of application domains needed to support manufacturing industry due to the inconsistency of concept semantics. Formal ontologies have shown promise in removing interpretation problems by computationally capturing the semantics of concepts, ensuring their consistency and thus providing a verifiable and shared understanding across multiple domains. The research work reported in this paper contributes to the development of formal reference ontology for manufacturing, which is envisaged as a key component in future interoperable manufacturing systems. A set of core manufacturing concepts are identified and their semantics have been captured in formal logic based on exploiting and extending existing standards' definitions, where possible combined with an industrial investigation of the concepts required. A successful experimental investigation has been conducted to verify the application of the ontology based on the interaction between concepts in the design and manufacturing domains of an aerospace component.

Keywords: ontologies; manufacturing ontology; semantics; core concepts; interoperability; formal ontology

1. Introduction

Modern manufacturing industries employ multiple Information and Communication Technology (ICT) tools for different domains. Effective communication across such domains is a key to competitiveness. This leads to the concept of 'interoperability', which is the ability of two or more systems or their components to seamlessly exchange knowledge (IEEE Std 1999; ISO/IEC-TR-10000-1 1998). Problems in interoperability have been estimated to cost billions of dollars annually (Brunnermeier and Martin 1999; Gallaher et al. 2009). The differences in the required terms and their semantics (intent and meaning) create interoperability problems for different ICT systems and domains. Nearly 70% of interoperability project costs are spent on identifying and reconciling the differences in the meanings of concepts i.e. semantics mismatches (Bussler et al. 2005). However, Semantic mismatches can be reconciled through the use of common semantics (Hakimpour 2003).

The international standards community has been providing a basis for information sharing through standards. Examples of such standards are the Initial Graphics Exchange Specification (Nagel, Braithwaite, and Kennicott 1980) standard for exchanging information between different Computer-Aided Design (CAD) applications, Parts Library (ISO-13584 2001), industrial manufacturing management data MANDATE standard (ISO 15531 2004), STEP standard (ISO-10303 2006) and its various parts. The concepts and structures provided by these standards are useful, but within their own narrow manufacturing domains, where meanings of concepts are understood by the community concerned. However, they have interpretation issues when it comes to using them in ICT systems across multiple manufacturing domains. For example, the definition of the concept 'component' varies not only across different standards but also across different parts of the same standard:

- 'Component' in ISO-10303-1 'A product that is not subject to decomposition from the perspective of a specific application'
- 'Component' in ISO-10303-224-'The component specifies either a Single_piece_part or another Manufactured_assembly used to define an assembly'

- ‘Component’ in ISO 19439:2006. [general] ‘Entity that is part of, or capable of becoming part of, a larger whole’

The understanding of the meaning of ‘component’ in the above example is down to the human interpretation of the text. They might be alternative definitions that are incompatible or they might be explaining the same concept but in different ways. It is not easy, if at all possible, to categorically identify the level of compatibility that exists across these definitions. Standards typically provide text-based informal definitions which are inconsistent across different standards and also across different parts of the same standard (Usman et al. 2011). Such text-based and inconsistent semantics can lead to multiple and possibly inappropriate interpretations when multiple manufacturing domains are using standards. Such problems can be resolved when humans are interacting with each other, however, this is difficult for modern manufacturing industries that use multiple ICT systems for multiple domains. Semantic differences across different ICT systems and domains can be reconciled through the use of computationally understandable (formal) common semantics and ontologies are a way of providing that.

Researchers over the years have contributed to the development of ontologies for manufacturing. These ontologies have ranged between informal (not understandable or readable by computers, e.g. a set of concepts on a paper), semi-formal (computers can read the concepts but do not understand their meanings) and formal (computers can under- stand i.e. interpret and act upon the meanings of concepts). A review of literature as well as the industrial investigation was undertaken to elicit the research gaps in present manufacturing ontologies. Particular research gaps being addressed in this paper are as follows: (i) There is a need to define a core set of manufacturing concepts to provide a common understanding for various strands of manufacturing (Chen, Doumeingts, and Vernadat 2008; Lee and Suh 2008); (ii) it was identified that the core manufacturing concepts should be formally defined to enable computer system to understand the semantics of concepts (Anjum et al. 2012; Chungoora et al. 2010); (iii) there is a need to define a manufacturing reference ontology (MRO) that can support the development of semantically sound application-specific ontologies for product design and production domains (Anjum et al. 2012; Chungoora and Young 2010).

This paper proposes a MRO to address these research gaps, where the definitions of the concepts are formal and therefore interpretable by computer systems which can ensure semantic integrity and help to identify and resolve any potential misinterpretation problems. The proposed ontology supports the development of semantically sound application-specific ontologies for product design and production domains. The ontology development and verification method- ology is based on the guidelines provided by Blomqvist and Öhgren (2008) and Noy and McGuinness (2000). In the light of these guidelines, a set of experiments have been conducted to verify the research claims. The syntactic integrity of the ontology is verified by loading the ontology as part of a knowledge base. The semantics of concepts are then tested by asserting facts into the knowledge base that holds the ontology. Two types of facts are asserted; firstly, those that are known to violate the formal definitions of concepts and then those which conform to formal definition. The results verify the capture of semantics. The proposed ontology is used to develop semantically sound application-specific ontologies for product design and production domains. The syntactic and semantic integrity of the application-specific domain ontologies are tested in the same way as the MRO is tested.

The paper is organised as follows. Section 2 provides a review of the relevant manufacturing ontologies and related research work. Section 3 outlines the requirements of the MRO identified as a result of the review. Section 4 presents an overview of the research approach to develop the proposed MRO. Section 5 presents the exploration of core manufacturing concepts, their textual definitions and their hierarchical model. Section 6 presents the process of formalisation of the semantics of concepts with examples. Section 7 presents the experimental investigation of the ontology and research claims. Section 8 presents the discussion and conclusions.

2. Previous work

The focus of this work is placed mainly on the production domain and partially on the design domain. The production domain is considered because it accounts for the cost of manufacture and design domain because it has a major influence on the majority of the costs (True and Izzi 2002).

Ontologies with all the concepts dependent on a certain domain are called domain ontologies (Borgo and Leitão 2007). Different domains require different domain ontologies. However, there is little interoperability between them due to the differences in their specific semantics. The computational communication between such domains will remain limited unless an understanding of the relationships that exist between them can be developed. This can be achieved through a reference ontology.

Foundation ontologies, which are formal ontologies developed with a view to cover the basic semantics of everything, can potentially act as reference ontologies for multiple domains. The intent to cover the semantics of everything makes concepts in foundation ontologies very generic, e.g. Particular, Endurant, Perdurant (from DOLCE (Masolo et al. 2003)), Abstract Entity and Concrete Entity (from ULO (IODE 2010)). This generic nature makes foundation ontologies applicable to any domain. For instance, the foundation concept ‘Resource’ may refer to *Machine Tools*, *Cutting Tools*, *Operators*, *CAD Tool*, *Printer*, *Time and Space etc*. This implies that a huge set of concepts which belong to different domains and which are semantically different may well be classed under a single foundation concept.

However, the same generic nature of foundation concepts makes foundation ontologies overly generic and wide-ranging for interoperability across specific domains (Borgo and Leitão 2007) such as product design and production. For example, a huge set of concepts classed as *Resource* would lead to the identification of similarities between vastly different concepts when viewed from the level of product design and production. This can create ambiguities for knowledge sharing across specific product design and production domains. So, a more specialised set of concepts with their semantics closer to the product design and production domains are required to support the semantic coherence i.e. consistency of interpreted meanings, between them. Using a foundation concept as a semantic base for product design and production domains will group together concepts that are vastly different at the application- specific levels. Such unwanted similarities or inability to distinguish semantically different concepts will create problems for interoperability across specific domains. Therefore, an ontology that is more specific than foundation ontologies and more generic than domain ontologies is required. Such an ontology is called a core ontology (Gangemi and Borgo 2004). Core ontologies provide a set of generic concepts whose semantics are shared across multiple, but not all, domains (Deshayes, Foufou, and Gruninger 2007). Therefore, a core manufacturing ontology should act as a reference for multiple manufacturing domains.

Researchers over the years have been trying to develop such reference manufacturing ontologies. As a result, a number of relevant models, hierarchies and frameworks, which can be considered as ontologies, have been developed. A list of such relevant ontologies is provided in Table 1 with the most relevant references being highlighted using bold text. A brief review of the most relevant research works is provided next.

Standards provide a number of useful concepts and textual definitions for developing a core formal manufacturing ontology (Young et al. 2007). Examples of such concepts are *Manufacturing Feature*, *Part, Resource*, *Process, Component, Manufacturing Method, Process Plan* etc. However, the concepts in most of these standards have text-based non-formal definition of concepts and the definitions of concepts vary not only across different standards but even across different parts of the same standard (Usman et al. 2011).

The issue of consistent semantic interpretation also exists with other non and semi-formal manufacturing ontologies. Examples of such ontologies are MOSES ontology (Molina and Bell 1999), CIMOSA ontology (CIMOSA-Association 1996; ESPRIT-Consortium-AMICE 1993), FDM ontology (Harding and Yu 1999), product ontology for integrating production planning and design (Tursi et al. 2009), a model to share manufacturing best practice knowledge (Gunendran and Young 2010), core product model (Fenves et al. 2008) and ontology-based tool for product data exchange (Chang Sahin, Terpenny 2008). However, these ontologies provide concepts and relations with textual definitions that can still be useful in defining a formal manufacturing ontology.

Researchers have tried to overcome the limitations of non-formal or semi-formal ontologies through the use of formal ontologies. Leimagnan et al. (2006) developed manufacturing semantics ontology (MASON) to formally capture the semantics of concepts related to manufacturing industries. The semantics were captured in formal logic using the web ontology language (OWL). Using MASON, the production knowledge related to individual operations could be formally captured. MASON can potentially facilitate in developing an understanding of the concepts and relations for the researched manufacturing ontology. MASON, however, does not consider the aspect of semantic coherence between different manufacturing domains and is limited by the low expressive power of OWL.

Existing formal manufacturing ontologies do not consider the design aspect in their ontologies and tend to focus on narrow domains within production e.g. production planning and scheduling (Borgo and Leitão 2007) and machining (Semere, Dislshad, and Bengt 2007). Some formal ontology research has considered interoperability between design and production and provided some useful input to understanding the requirements. For example, enabling translation of CAD models to CAPP models (Dartigues et al. 2007), attempting to share welding knowledge with design (Chang, Rai, and Terpenny 2010), enforcing the use of predefined features on designers (Chen and Wei 1997; Hoque and Szecsi 2008) and providing a framework for manufacturing (Chungoora 2010; Lee and Suh 2008).

Table 1. List of ontologies shortlisted from the reviewed ontologies.

Sr. No.	Authors	Year	Ontology	Relevance to the MCCO	Formality
1	Gunendran and Young	2010	Capturing best practice manufacturing Knowledge	Very High	Semi formal
2	Chang, Rai, and Terpenny	2010	Design for manufacture (DFM) ontology	High	Fomral
3	Chungoora	2010	Semantic interoperability framework	Very High	Formal
4	Chen, Chen, and Chu	2009	Development of a mechanism for ontology-based product lifecycle knowledge integration	Medium	Semi formal
5	Tursi et al.	2009	Product ontology for embedding product information for interoperablility	Medium to High	Semi formal
6	Dartigues et al.	2007	Ontology of features to support CAD/CAPP integration	Very High	Formal
7	Blomqvist and Öhgren	2008	Ontology for model-based software engineering of dependable systems (SEMCO) project	Medium	Semi formal
8	Chang, Sahin, and Terpenny	2008	An ontology-based support for product conceptual design	High	Semi formal
9	Lee and Suh	2008	ontology-based multilayer knowledge framework (OMKF)	Very High	Formal
10	Lagos and Setchi	2007	Manufacturing ontology for e-learning	Low	Semi formal
11	Abdul-Ghafour et al.	2007	A common design feature ontology for product data semantic interoperability	High	Formal
12	Terzi et al.	2007	Holonic metamodel for product information traceability	High	LW
13	Semere, Dislshad, and Bengt	2007	Machining ontology	very High	Formal
14	Borgo and Leitão	2007	ADAptive holonic control architecture for distributed manufacturing systems is the architecture (ADACOR), core manufacturing ontology Core product model (CPM)	Very High	Formal
15	Fenves et al.	2008		High	LW
16	Leimagnan et al.	2006	MAnufacturing semantics ontololgy (MASON)	High	Formal
17	Sarder	2006	Design ontology	Medium to High	Formal
18	Patil, Dutta, and Sriram	2005	Product semantic representation	Medium	Formal
19	Harding, Popplewell, and Cook	2003	Manufacturing systems engineering (MSE) ontology	Medium	Formal
20	Schlenoff et al.	2000	Process specification language (PSL)-ISO-18629	Medium	Formal
21	CIMOA Architecture	1996	Computer-integrated manufacturing open system architecture (CIMOSA) ontology	Medium	LW
22	Harding and Yu	1999	Factory design model (FDM)	Medium	Semi formal
23	Molina and Bell	1999	Manufacturing model for (MOSES) project	High	Semi formal
24	Uschold et al.	1998	Enterprise project	Medium	Semi formal
25	Fox and Gruninger	1997	TOVE project	Medium	Formal
26	ISO Standards		ISO-10303-1,49, 224, and 239, ISO-13584, ISO-15531-1 (MANDATE), ISO19,439, ISO/IEC-TR-10000-1, ISO/ TS-10303-1017, 1018, 1022, 1164, ISO-13399, ISO-15,531-43,	Medium to Very High	Semi formal

3. Requirements of a reference ontology for manufacturing

The requirements have been identified with a view to the development of a manufacturing ontology. The process used to identify the requirements is based on the guidelines provided by Blomqvist and Öhgren (2008) and Noy and McGuinness (2000) in their ontology development methodologies.

Over the years, researchers and engineers working in the area of manufacturing knowledge management have been dealing with the sets of information relating to manufacturing resources (Leimagnan et al. 2006; Molina and Bell 1999; Vichare et al. 2009), manufacturing processes (Feng and Song 2003; ISO-18629 2004; ISO-18629-1 2004; Todd 1994), manufacturing facilities (Lin and Harding 2007; Simpson, Hocken, and Albus 1982; Zhao et al. 1999), features (Dartigues et al. 2007; Fenves et al. 2008), part family models (Chungoora and Young 2010; Young et al. 2007) and manufacturing methods and process plans (Gunendran and Young 2010), etc. This shows that the manufacturing knowledge has traditionally been grouped under different categories. This highlights the need to identify such concepts in the manufacturing ontology that relate to different categories of manufacturing knowledge.

An industrial study was carried out with an aero engine company. This study provided an industrial foundation for the proposed MRO. The design and manufacturing of an aero engine disc was studied in detail. The study was directed at identifying and understanding the core concepts required for a manufacturing ontology. It was also directed at understanding the limitations of present methods of interoperability between design and production. The study was fundamental in understanding the issues of communication between design and production, defining several core concepts and building application-specific ontologies. Based on the investigation of the problem through the relevant literature and the industrial study, it was identified that the present manufacturing ontologies need to be further developed and extended. It was identified that a core set of manufacturing concepts need to be defined for manufacturing industry (Lee and Suh 2008). It was identified that a set of core manufacturing concepts and relations should be formally defined to support interoperability across specific product design and production domains (Anjum et al. 2012; Chen, Doumeingts, and Vernadat 2008; 2008; Chungoora and Young 2010). The requirements can be summarised as follows: (1) there is a need to define and categorise manufacturing concepts in the light of the previous research work on manufacturing ontologies, (2) there is need to define a core set of manufacturing concepts, which are neither as generic as foundation ontologies nor as specific as application-specific ontologies, (3) there is a need to formally define the core manufacturing concepts to enable computer systems to interpret the semantics of the concepts and (4) the core manufacturing ontology should support the development of application-specific ontologies in product design and production domains. The main research questions in this regard, which also act as the competency questions for the ontology are:

(1) What are the core set of manufacturing concepts, relations and concept categories that sit between the foundation concepts and application-specific manufacturing concepts?
(2) Can the formally defined set of manufacturing concepts enable computer systems to interpret the meanings of the concepts and hence respond accordingly?
(3) Can the formally defined set of concepts provide an ability to computer system to report the violations of the formal definitions and guide the user to the requirements that need to be met to satisfy the formal definitions?
(4) Can a formally defined core manufacturing ontology support the development of semantically sound application-specific ontologies for product design and production domains?

A method to devise a formal manufacturing ontology that can answer the above questions and thereby fill the research gap has been devised.

4. An overview of the research concept

An overview of the proposed research methodology to answer the research questions raised in Section 3 and fill the research gaps is presented in Figure 1. The portion A in the Figure 1 shows the ontology development methodology. There are several ontology development methodologies that are reported in literature, e.g. IDEF-5 methodology (IDEF5 Method Report 2005; METHONTOLOGY (Fernández-López 1999), CyC methodology (Elkin and Greiner 1993) and methodologies by Uschold and King (1995), Gruninger and Fox (1995), Swartout et al. (1997), Staab et al. (2001) and Corcho, Fernandez-Lopez, and Gomez-Perez (2003). However, for the development of a formal MRO, Blomqvist and Öhgren (2008) and Noy and McGuinness (2000) both appear to offer potentially useful methods when applied to manufacturing. Their methodologies have been used, with some adaptation with the additional steps of formally defining the concepts and testing their semantics (step 4.3 and step 5 in Figure 1).

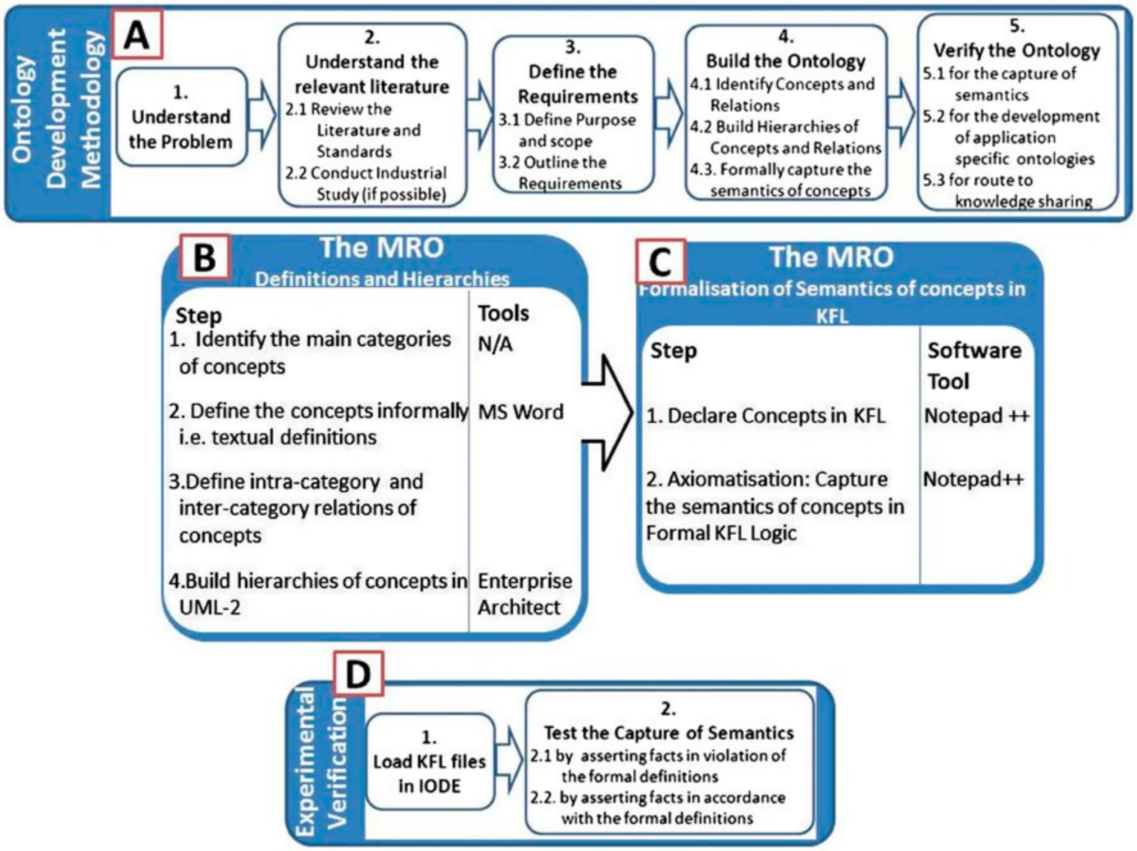

Ontology Development Methodology
A
1. Understand the Problem
2. Understand the relevant literature
2.1 Review the Literature and Standards
2.2 Conduct Industrial Study (if possible)
3. Define the Requirements
3.1 Define Purpose and scope
3.2 Outline the Requirements
4. Build the Ontology
4.1 Identify Concepts and Relations
4.2 Build Hierarchies of Concepts and Relations
4.3. Formally capture the semantics of concepts
5. Verify the Ontology
5.1 for the capture of semantics
5.2 for the development of application specific ontologies
5.3 for route to knowledge sharing
B
The MRO
Definitions and Hierarchies
Step
Tools
1. Identify the main categories of concepts
N/A
2. Define the concepts informally i.e. textual definitions
MS Word
3.Define intra-category and inter-category relations of concepts
4.Build hierarchies of concepts in UML-2
Enterprise Architect
C
The MRO
Formalisation of Semantics of concepts in KFL
Step
Software Tool
1. Declare Concepts in KFL
Notepad ++
2. Axiomatisation: Capture the semantics of concepts in Formal KFL Logic
Notepad++
D
Experimental Verification
1. Load KFL files in IODE
2. Test the Capture of Semantics
2.1 by asserting facts in violation of the formal definitions
2.2. by asserting facts in accordance with the formal definitions

Figure 1. An overview of the proposed research concept.

The initial steps in the adapted methodology, as shown in Figure 1 and is to build an understanding of the problem through general awareness, reviewing relevant literature and performing industrial studies. The third step is to define the requirements of the ontology clearly based on the research gaps identified through the literature and the industrial study. The fourth step is to build the ontology. This step requires significant research effort to clearly identify the concepts and relations. The structure of the ontology follows by defining (1) categories of concepts, (2) concepts belonging to each category and (3) the relations between concepts. The full development of the concepts leads to their formal definitions. After the hierarchical structures of the concepts and their non-formal semantics have been defined, the next step, 4.3, is the formal capture of the semantics of each concept. This step takes a non-formal ontology through to a formal one by adding logical axioms that constrain the meanings of the concepts. The last step, i.e. step 5, is the verification of the researched ontology against the requirements through experimentation.

Portion [B] in Figure 1 represents the method for the development of the ontology hierarchy and the non-formal definition of the concepts. This consists of four main steps. Portion [C] shows the method of formally capturing the semantics by: (1) declaring them in a formal logical language and (2) writing formal axioms to capture the semantics. Portion [D] shows the verification process of the MRO. The MRO is experimentally investigated for the capture of semantics and for supporting the development of design and production ontologies by: (1) asserting facts in violation of the formal definition and then (2) asserting facts in accordance with the formal definitions.

5. Development of the MRO

5.1 *The core concepts and relationships for the MRO*

After the requirements of the MRO have been identified, the next step is to synthesise the understanding gained from the work reported in Table 1 and the industrial study into a core set of manufacturing concepts and relations. It is understood that manufacturing industry requires information from several different sources and the different sources of information have been grouped into different categories based on the research work. As mentioned in Section 3, manufacturing information and knowledge is classified into several different sets. The industrial investigation that was undertaken as part of this research identified the need for multiple categories of information. This investigation, along with the review of literature, led to the proposal that the manufacturing concepts should be categorised into eight main concept areas. These being, *Realised Part, Part Version, Manufacturing Facility, Manufacturing Resource, Manufacturing Method*, *Manufacturing Process*, *Feature* and *Part Family*. The eight categories are not claimed to be complete but they provide a step towards a complete MRO. In the context of providing core manufacturing concepts to support the development of product design and production domain ontologies, the eight categories of concepts are sufficient. These categories are not independent of each other and relations exist between concepts from each category (intra-category relations) and between concepts from different categories (inter-category relations). The resulting concepts and relations have been modelled in Unified Modelling Language-2 (UML-2) as a semi-formal representation of the ontology as shown in Figure 2. The concepts in larger boxes in italic bold text represent the main categories of concepts and the bold lines represent the inter-category relations. The thinner lines represent the intra-category relations. The concepts in normal size boxes and text are intra-category concepts.

The different shades of boxes in Figure 2 are used to distinguish the concepts that are (1-Grey) original MRO concepts, (2-White) concepts adapted from others and (3-black) concepts adopted from others. The definitions of the original MRO concepts are also based on the understanding gained from the literature review and the industrial study. For example, the concept *ProductionPart* is based on the understanding gained from ISO-15531-1 (2004) and ISO/TS-15926-4 (2007). The adopted concepts have been included along with their definitions from their different sources, e.g. the concepts *ManufacturingProcess* has been adopted from ISO-15531-1 (2004). Similarly, all *ManufacturingFacility* concepts are adopted from Molina and Bell (1999) and Zhao et al. (1999). The definition of the *RealisedPart* has been adopted from the concept *RealisedProduct* from ISO/TS-10303-1164 2004. The adapted concepts are the ones whose definitions have been slightly altered from the original definitions from one or more sources. For example, definition of the concept *PartFamily* as ‘a parametric *Part* which represents a *Family* of *Part*s which have the same *PartFamily*

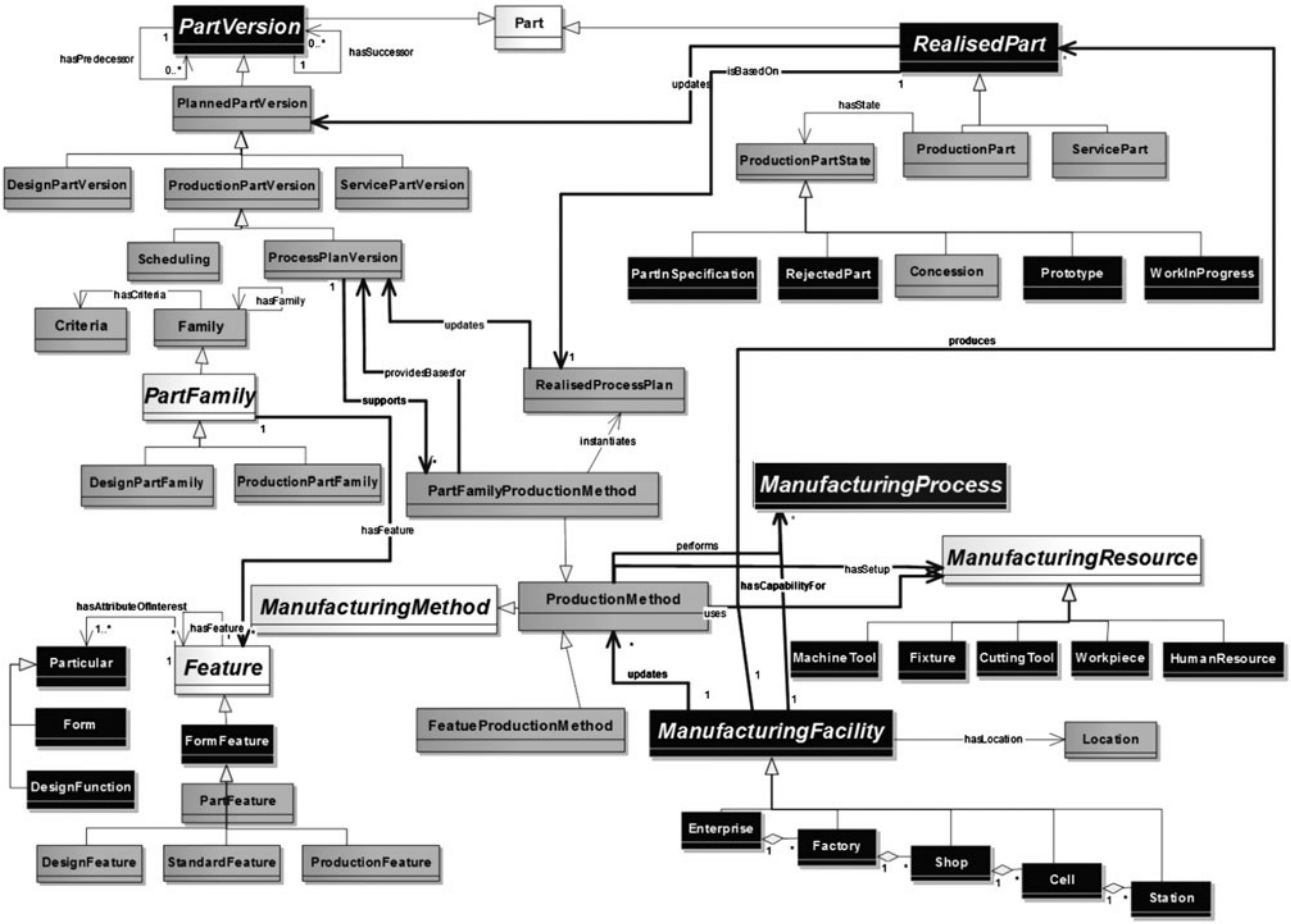


Figure 2. Semi-formal diagrammatic representation of MRO.

criteria' is adopted from Gunendran and Young (2010) and Chungoora (2010). Similarly, the concept *ProductionPart* is adopted from ISO-15531-1 2004 and ISO/TS-15926-4 2007.

One of the main activities in the ontology development has been to elicit key terms in an attempt to create the backbone of concepts for each category in the MRO. A discussion of all the concepts is not possible in this paper; therefore concepts from '*Feature*' category have been selected to illustrate the method.

5.1.1 ***Feature concepts and relations***

From feature-based CAD modelling to the definition of machining and assembly features, the concept of *Feature* has significant importance in different manufacturing domains. In the manufacturing ontology research, *Feature* concepts are one of the most explored and referenced concepts (Abdul-Ghafour et al. 2011; Chungoora 2010; Dartigues et al. 2007; Fenves et al. 2008; Gunendran and Young 2010). *Feature* concepts have been selected to explain the exploration of concepts because they are one of the main sets of concepts in the MRO. They are relatively simple to explain with simpler axioms. *Feature* concepts act as building blocks for the design and production part families. They bring together manufacturing information from different categories.

At the generic level, *Feature* can be interpreted independently of any viewpoint. In that context, the definition of *Feature* as 'a prominent attribute or aspect of something (Webster's online Dictionary 2011; WordNet 2010)' establishes the essence of a *Feature* to have an attribute of interest. According to this, *Feature* is a very generic term which can be applied to many areas outside of manufacturing and also in many different ways within manufacture, such as the feed and speed of a machine, the rendering ability of CAD software, the cycle time of an operation colour of a part, size and shape of an article, etc. Therefore, the concept *Feature* at a generic level has been informally defined as: 'Feature is anything having a particular attribute of interest'

The UML diagram in Figure 2 portion [7] shows the semi-formal representation of the concept *Feature* by showing that *Feature* is related to *Particular* by relation *hasAttributeOfInterest*. This shows that a *Feature* has a *Particular* attribute of interest. The attribute[1] of interest of a *Feature* can be either an event or an object.

Features in manufacturing will always have a form, so this requires the capture of semantics involving an associated form. To do this, a more specialised concept *FormFeature*, is identified. Examples of *FormFeature* can be key way slot, milling slot, drain hole, drilling hole, etc. The concept *FormFeature* is also found in previous literature. Semere, Dislshad, and Bengt (2007), for example, had *FormFeature* as one of the main concepts in their machining ontology, where it represented a manufacturing product with 'geometry', 'tolerances' and 'other technical data' as the required attributes. *FormFeature* was used in the core product model (CPM) (Fenves et al. 2008) to capture geometry and material. In this paper, *Form Feature* is required to capture the form of features irrespective of the material. This helps in keeping the core concepts flexible for use by application-specific ontologies. Therefore, *FormFeature* is informally defined as 'a *Feature* which has a *Form* as its required attribute of interest'. This definition seems to be common sense-based and simplistic, however, the real value comes from the formalisation of this definition in logic, which provides a computer-interpretable definition.

The semi-formal diagrammatic view of the semantics of *FormFeature* is shown in Figure 2. The *Feature* concept subsumes the *Form Feature* concept (Figure 2). The relation *hasAttributeOfInterest* relates a *FormFeature* to its *Form*. This relation is inherited from '*Feature*' and, therefore, does not need to be shown with *FormFeature*.

In the context of parts, the concept *FormFeature* should represent a certain portion on a part, e.g. groove on a disc and hole in a pipe. Therefore, a concept *PartFeature* is identified to capture the semantics that refer to the part on which the *FormFeature* exists. A *PartFeature* may exist in different manufacturing domains; therefore, its semantics should be generic enough to suit any of those domains. In this context, a number of definitions of *Feature* are available in international standards (ISO-10303-APs1101, 1130, 207, ISO 13,584). Based on these definitions and the industrial study, *Part Feature* is informally defined as 'a *FormFeature* associated to a *Part*'.

The concept *PartFeature* is significant for both product design and production domains. Therefore, concepts capturing the design and production semantics for features need to be defined. The concepts identified in this regard are *DesignFeature* and *ProductionFeature*. The identification of a *DesignFeature* is driven by the functional requirements, e.g. *Balancing Feature* and *StressReducingFeature*. Therefore, *DesignFeature* is defined as 'a *PartFeature* having a *DesignFunction* as a defining attribute of interest'.

The concept *ProductionFeature* is informally defined based on the understanding gained from the industrial study and the ISO standard (ISO-10303-224) as 'a *PartFeature* having a *ProductionMethod* as a defining attribute of interest'. Similarly, concepts from other categories have been defined informally. ISO standards have been useful, in particular, in providing an understanding of several MRO concepts. For example, definition of concept *RealisedPart*, which is required to capture the physically present parts produced as a result of some manufacturing activity, is adopted from the concept *Realised Product* from the STEP standard (ISO/TS-10303-1164 2004). The informal definition of *RealisedPart* is ‘a part that exists physically in the real world and whose properties can only be known by observation’.

Similarly, concept *PartVersion,* which is required to capture historical versions of the parts, is defined based on (ISO/TS-10303-1022 2004) as ‘a *PartVersion* refers to a version of part which captures the history of the part’.

In parallel to the identification of concepts goes the process of defining the relations between concepts. Relations are required to support the understanding of the meanings and facilitate the development of the structure of the manufacturing ontology. For example, with every new *RealisedPart* some new knowledge is generated, which updates the historical knowledge which is captured through *PartVersion*. Therefore, a relation *updates* is defined to relate the concept *RealisedPart* to *Part Version*. In this way, relations are defined throughout the ontology to represent a coherent and comprehensive, yet generic manufacturing structure.

The model presented in Figure 2 is semi-formal. In order for the computer system to understand the semantics and respond accordingly, the semantics should be captured in formal logic. The language used for that purpose in this research is the knowledge frame language (KFL), which is based on the international standard for logic-based formalism ISO/IEC-24707 (2007). The next section presents an introduction to KFL and explains why this was chosen to formalise the ontology.

6. Formalisation of the ontology

Before explaining the formalisation of the ontology, it is important to explain why KFL has been chosen as the ontology development language.

6.1 Why Knowledge Frame Language (KFL)

Formal ontology development languages are all based on sub-sets of first-order logic. These can be broadly classified as description logic based languages, e.g. resource description framework (RDF/RDF(S)) and web ontology language (OWL) and common logic based languages, e.g. common logic interchange format (CLIF), knowledge interchange format(KIF), knowledge frame language (KFL). Although RDF and OWL are more widely used, common logic based languages have more expressive power and provide better inference capabilities, which are better suited to the complexities of manufacturing concepts and relations (Chungoora 2010).

The development of the MRO involves formalisation of definitions of concepts which requires high expressive and inference power. Therefore, the choice had to be made between CLIF, KIF, KFL and other languages in the same category. KFL was chosen because it is the coding language for integrated ontology development environment (IODE), which is a major commercially available ontology development environment. Moreover, KFL is based on common logic, which is an international standard for first-order logic formalisms (ISO/IEC-24707 2007).

KFL is developed by Highfleet Inc. that provides the syntax and formalisms required to develop complex formal ontologies. The ability to encode ontological content in KFL derives from Highfleets upper level ontology (ULO), which provides the basic concepts and constructs for building ontologies. The next section explains the use of KFL to formalise the MRO with the help of an example.

The lightweight formalisation of the MCCO presented in Sections 5.2, 5.3 and 5.4 is part of the overall formalisation of the MCCO. The KB system can only understand the meaning of concepts and identify the similarities between product design and production concepts after the heavyweight formalisation of the semantics of concepts has been defined.

The process of formalisation in KFL consists of two steps: (1) the declaration of the required concepts, relations and functions and (2) axiomatisation to formally capture the semantic constraints that apply to the concepts. The concept *Feature* is used to illustrate the formal capture of semantics using KFL in the next section.

6.2 Declaring concepts, relations and functions

The concepts in KFL are called ‘Properties’ and are declared using the directive ‘:Prop’. For example, the first step to formalise *Feature* is to declare this concept in KFL as follows:

```
:Prop Feature

:Inst Type

:sup Object
```

The directive ':Inst' is used to declare the type of instantiation of concepts. Mostly the instantiations are either of kind '*Type*' or '*MaterialRole*' in KFL. Type represents the properties that permanently 'stick' to their instances. For example, an instance of a property *Person* as type will always remain a person. *MaterialRole* in contrast, can come and go, e.g. an instance of a *Person* as a *MaterialRole* can be a Student at one time and a Teacher at another. The directive ':sup' specifies the parent concept of a concept and thus builds a hierarchy.

After declaring the concepts, the required relations are declared. For example, the relation '*hasAttributeOfInterest*' is required to formalise the semantics of *Feature,* which is declared as follows;

:Rel has Attribute Of Interest

:Inst Binary Rel

:Sig Feature Particular

According to the definition, a *Feature* must have an attribute of interest. The association of attribute of interest to *Feature* is captured through the relation *has AttributeOfInterest*. The directive ':Rel' is used to declare the relations. The directive ':Inst' specifies the kind of relation and the number of the concepts involved, e.g. a ':Inst BinaryRel' represents a relation over two concepts. The concepts involved in a relation are specified using the 'sig- nature' directive, i.e. ':Sig', e.g. the concepts involved in the relation *has AttributeOfInterest* are *Feature* and *Particular* as represented by the directive:Sig. In the relation *hasAttributeOfInterest, Particular* is a foundation concept from the ULO which can represent both events and objects. The concept *Particular* has been used because the attribute of interest of a *Feature* can be an object as well as an event and *Particular* provides both of these required semantics.

6.3 Axiomatisation

After the declaration of concepts and relations, the ontology still remains semi-formal. It becomes formal through axiomatisation. Axiomatisation refers to the defining of axioms to formally constrain the interpretations of the concepts. This equips the system with an ability to understand the semantics of concepts and thus helps in solving the semantic interoperability issues. Axioms in the context of KFL can either be 'inference rules' or 'integrity constraints (ICs)'. The inference rules, as the name suggests, are used to make inferences, whereas the ICs formally capture and constrain the semantics of concepts. For example, to capture the definition of *Feature* formally, the following IC is written:

```
(=> (Feature ?feature)
       (exists(?p)
   (and (Particular ?p)
       (hasAttributeOfInterest ?feature ?p))))
:IC hard 'Feature requires an attribute of interest'
```

The above axiom formally states 'If there is a *Feature* "?feature" then a *Particular* "?p" related to that *Feature* through the relations *hasAttributeOfInterest* has to exist'. This implies that in order to assert a *Feature* in the KB, its *Particular* attribute of interest had to be defined and related to it. The directive ':IC hard' at the end of the axiom specifies that the IC cannot be violated during the assertion of facts and that the asserted facts have to satisfy the condition specified by the IC message. The firing of the ICs when the definition of a concept is violated will show that the system understands the semantics of the concepts and the message in the ICs will direct the user to the requirements that need to be satisfied. This is experimentally investigated in Section 7.

In a similar way, the semantics of other *Feature* concepts i.e. *FormFeature*, *PartFeature, ProductionFeature* and *DesignFeature* are also captured. These axioms are shown below:

FormFeature 'a *Feature* which has a *Form* as its required attribute of interest'

```
(=> (FormFeature ?ffeature)
    (exists (?form)
        (and (Form ?form)
            (hasAttributeOfInterest ?ffeature ?form))))
:IC hard ‘Every FormFeature has a form’
```

DesignFeature ‘a *PartFeature* having a *DesignFunction* as a defining attribute of interest’.

```
(=> (DesignFeature ?df)
    (exists (?dfunction)
        (and (DesignFunction ?dfunction)
        (hasAttributeOfInterest ?df ?dfunction))))
:IC hard ‘A DesignFunction has to be defined as an Attribute of Interest for the DesignFeature’
```

ProductionFeature ‘a *PartFeature* having a *ProductionMethod* as its defining attribute of interest’

```
(=> (ProductionFeature ?pf)
    (exists (?fpm)
        (and (FeatureProductionMethod ?fpm)
            (hasAttributeOfInterest ?pf ?fpm))))
:IC hard ‘A ProductionMethod must be defined for the asserted ProductionFeature’
```

PartFeature ‘a *FormFeature* associated to a *Part*.’

```
(=> (PartFeature ?Pfeature)
    (exists (?P)
    (and (Part ?P)
        (associatedTo ?Pfeature ?P))))
:IC soft ‘Every Part feature has an associated Part’
```

Note: the formal definition of *PartFeature* is mentioned last because to put in a sequence the concepts having the relation *hasAttributeOfInterest. PartFeature* includes an *associatedTo* relation and is therefore separated.

The formalisation of the semantics of *Feature* concepts is presented as an example and the same methodology applies to the formalisation of semantics of all other MRO concepts. This formalisation equips the computer system with an ability to interpret the meanings of concepts. The system can understand the semantics and can respond in accordance with the formally captured semantics. The system can also direct the user to the requirements that need to be met for satisfying the formal semantics. The fully formalised MRO provides a core ontology as a reference for the capture of production knowledge, building semantically sound application-specific domain ontologies. This is experimentally investigated in Section 7 of this paper.

7. Experimental verification

7.1 Experimental design

7.1.1 Experimental tools and languages

The tools used for experiments are briefly discussed here. The diagrammatic representation of the ontology, in the form of hierarchical models, was prepared in unified modelling language and the tool used was Enterprise Architect V-8. The UML models of the ontology are then coded in KFL to formalise the ontology. The tool for writing the KFL code was Notepad ++ (Notepad ++ 2012), which is an open source text editor that supports several formats. The KFL files of the ontologies are then loaded in the ontology development environment, which is integrated ontology development environment (IODE) developed by the Highfleet Inc. (Highfleet 2012). The ontology is loaded in IODE as part of the knowledge base (KB) termed eXtensibel Knowledge Server (XKS). The XKS knowledge bases are used in addition to IODE to serve three functions, i.e. (1) hold the ontology, (2) enable assertion of facts into the ontological knowledge base and (3) enable inferences to be made from the ontological knowledge base. These functionalities are used in experimentation.

7.1.2 Experimental set-up

For the experimentation, the following KFL files were prepared.

(1) Text files containing the MRO coded in KFL.
(2) Text file containing the application-specific product design ontology coded in KFL.
(3) Text file containing the application-specific production ontology coded in KFL.
(4) Text files containing KFL coded facts for different MRO concepts, for application-specific product design concepts and for application-specific production concepts.
(5) Text files containing KFL codes for making inferences in the ontology and knowledge base.

First, the KFL files containing the code for the MRO as well as the application-specific product design and production ontologies were loaded into the IODE as XKS knowledge bases. A successful loading of the KFL files in IODE ensures that there are no syntactic errors in the MRO as well as the application-specific ontologies. The

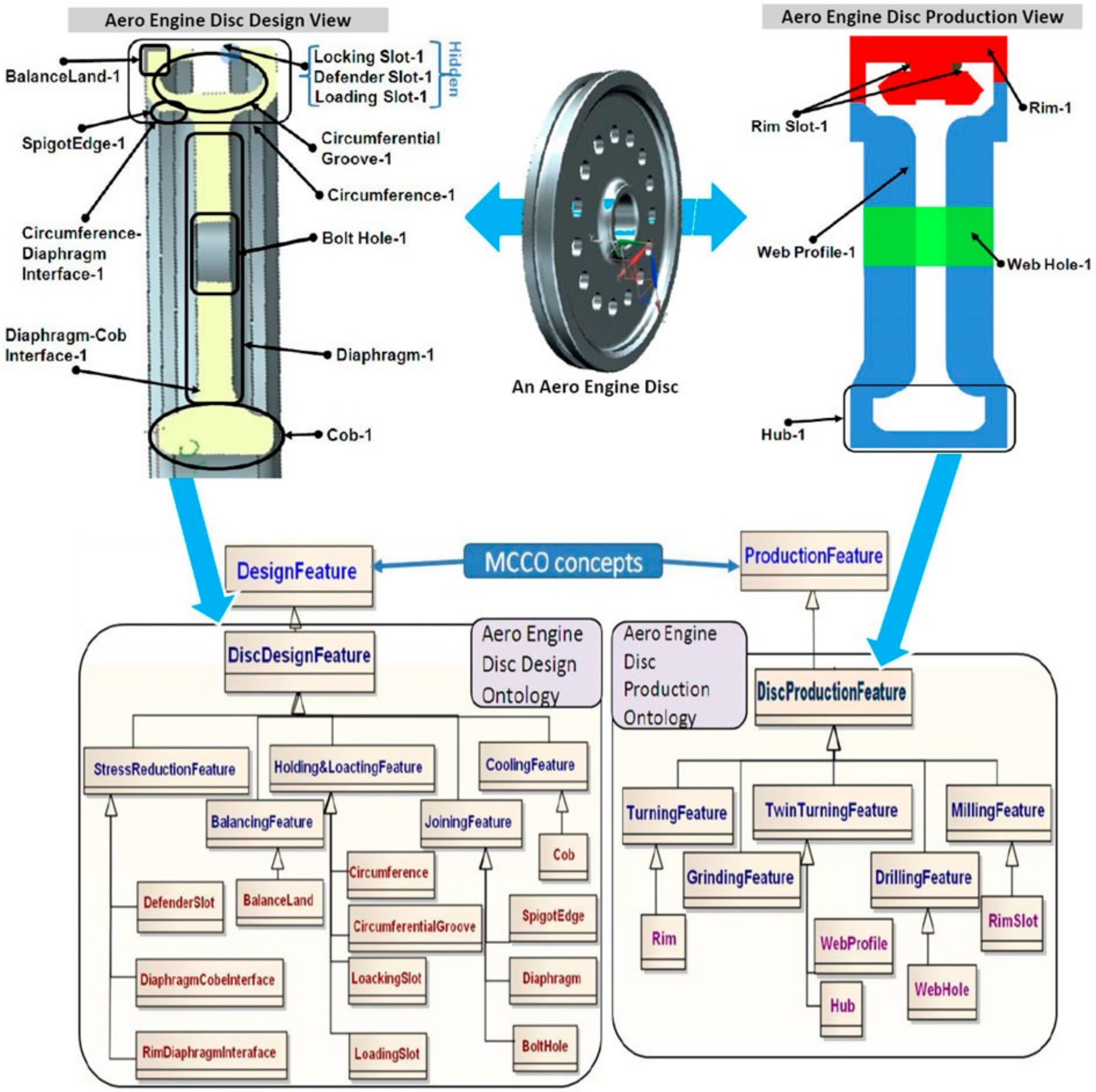


Figure 3. Nature of experiments to verify the development of application-specific design and production ontologies.

concepts for application-specific ontologies were identified based on the understanding gained from the industrial study. Because the study was in an aero engine disc, the application-specific ontologies relate to design and produc-tion of an aero engine disc. Figure 3 shows a view of the aero engine disc along with its design and production views. Figure 3 also shows the UML representation of those ontologies in the bottom half. Note that the ontologies are based on the MRO concepts of *DesignFeature* and *ProductionFeature*. The application-specific ontologies are named AeroEngineDiscDesignOntology and AeroEngineDiscProductionOntology. These ontologies were loaded in IODE as different XKS knowledge bases. They were then merged with the main MRO XKS and the set-up was ready for experimentation.

7.1.3 Nature of experiments

Two main sets of experiments were performed to (1) verify the capture of the semantics of concepts and (2) verify the ability of the MRO to support the development of semantically sound application-specific ontologies. The nature of experiments is briefly explained in the next section.

The verification of the semantics is done by asserting facts into the XKS. Facts are first asserted in violation of the formal definitions of concepts. The response of the system verifies if the system is able to identify the facts that violate the formal definitions. For example, if the facts that violate the formal definitions are successfully asserted, then the ontology does not understand the semantics, because either the semantics have not been captured or have been captured incorrectly. In such case, there will be a need to reformalise the ontology to capture the required semantics. If the ontology is able to identify the facts that violate the semantics and does not allow their assertion, this confirms the expected results. A set of experiments were then performed to check that the assertion of correct facts that conform to the formal definitions are accepted. This confirms that the semantics of the concepts have been interpreted correctly by the computer system. The experiment is detailed in Section 7.2.

The second set of experiments is more comprehensive. This involves the addition of experiments to verify the ability of the MRO to support the development of semantically sound application-specific ontologies for product design and production domains. The set-up for these experiments has been explained in Section 7.1.2. As mentioned, the two application-specific ontologies, based on industrially studied aero engine disc are, AeroEngineDiscDesignOntology and AeroEngineDiscProductionOntology. The concepts for these ontologies (shown in Figure 3) were coded in KFL and loaded into the system. The ability of the MRO to support the application-specific ontologies comes through the verification of the semantics as explained in Section 7.3.

7.2 Testing the capture of semantics

This experiment aims to achieve the following two objectives:

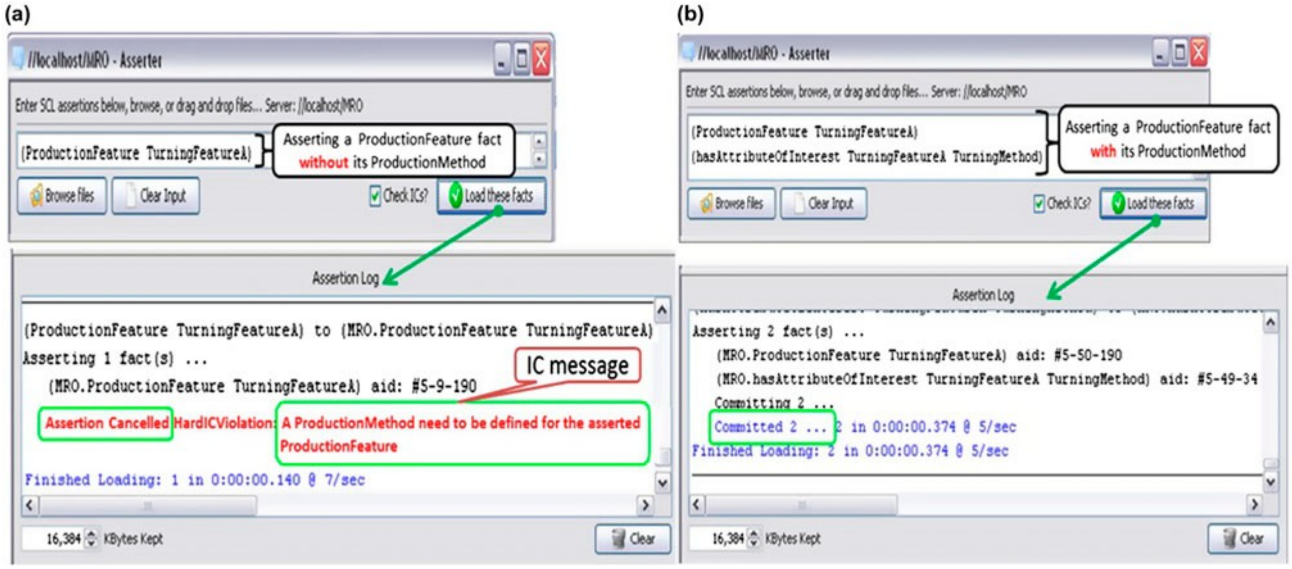
(1) To verify the formal capture of the semantics of the concepts.
(2) To confirm the ability of the system to understand the semantics and guide the user to the requirements that need to be met to satisfy the formal definition.

7.2.1 Procedure

This experiment is conducted using the *Feature* concepts. The concept *ProductionFeature* is selected because of the focus on production domain. The verification will come as results of the answers to the following questions.

(1) Does the system allow the assertion of facts that violate the formal definition of concept?
(2) Does the system report the violation of definitions when the violating facts are asserted?
(3) Does the system report the requirements that need to be met for a fact to be successfully asserted?
(4) Does the system allow the assertions when the facts satisfy the formal definitions of the concepts?

A fact for *ProductionFeature* is first asserted without a production method as its attribute of interest and then with a production method. Figure 4(a) shows the assertion of a *ProductionFeature*'s fact (TurningFeatureA) without its *ProductionMethod* and Figure 4(b) shows the assertion of the same fact with its *ProductionMethod* (TurningMethod) as its


(a)
//localhost/MRO - Asserter
Enter SQL assertions below, browse, or drag and drop files... Server: //localhost/MRO
Browse files
Clear Input
☑ Check ICs?
Load these facts
Assertion Log
16,384 KBytes Kept
Clear
(b)
//localhost/MRO - Asserter
Enter SQL assertions below, browse, or drag and drop files... Server: //localhost/MRO
Browse files
Clear Input
☑ Check ICs?
Load these facts
Assertion Log
16,384 KBytes Kept
Clear

```
(ProductionFeature TurningFeatureA)
```

Asserting a ProductionFeature fact without its ProductionMethod

```
(ProductionFeature TurningFeatureA) to (MRO.ProductionFeature TurningFeatureA)
Asserting 1 fact(s) ...
   (MRO.ProductionFeature TurningFeatureA) aid: #5-9-190
  Assertion Cancelled HardICViolation: A ProductionMethod need to be defined for the asserted
                                       ProductionFeature
Finished Loading: 1 in 0:00:00.140 @ 7/sec
```

IC message

```
(ProductionFeature TurningFeatureA)
(hasAttributeOfInterest TurningFeatureA TurningMethod)
```

Asserting a ProductionFeature fact with its ProductionMethod

```
Asserting 2 fact(s) ...
   (MRO.ProductionFeature TurningFeatureA) aid: #5-50-190
   (MRO.hasAttributeOfInterest TurningFeatureA TurningMethod) aid: #5-49-34
   Committing 2 ...
   Committed 2 ... 2 in 0:00:00.374 @ 5/sec
Finished Loading: 2 in 0:00:00.374 @ 5/sec
```

Figure 4. Asserting a ProductionFeature fact (a) without its ProductionMethod (b) with its ProductionMethod.

attribute of interest. In Figure 4(a) and (b), upper boxes are ‘fact assertion boxes’ and the lower boxes are the ‘Assertion Logs’ that show the results of assertions.

The above experiment only presents one example of verification of the captured semantics; similar experiments have been conducted for the other MRO concepts. A discussion on these results is as follows.

7.2.2 Discussion on results of the experiment

As shown in Figure 4(a), when a *ProductionFeature* fact was asserted without its *ProductionMethod*, the assertion was cancelled because of the violation of a hard IC. The IC was defined to formally capture the semantics of *ProductionFeature* and therefore not allow any assertions that violate the formal definition. According to the definition, a *ProductionMethod* should be defined for a *ProductionFeature* as its attribute of interest. Because in the assertion in Figure 4(a), no *ProductionMethod* was defined for the asserted *ProductionFeature*, the assertion was cancelled.

The cancellation of the assertion shows that the system has formally captured the semantics of *ProductionFeature*. It also shows that the system has been able to compare the asserted facts with the formal definition and has identified that the asserted fact is in violation of the formal definition. Not only that, the system also guided the user to the requirement that needs to be satisfied through the IC message which stated that ‘A ProductionMethod needs to be defined for the asserted ProductionFeature’. It shows that the system does not allow the assertions which violate the formal definitions and also specifies the requirements that need to be satisfied in accordance with the formal definitions of concepts.

On the other hand, when the same *ProductionFeature* fact ‘TurningFeatureA’ was asserted with its *ProductionMethod*, the assertion was a success and no ICs were violated (Figure 4(b)). The confirmation of assertion is done by the message stating ‘committed’ Figure 4(b).

Questions, such as ‘can a ProductionFeature have a name that is incompatible with the ProductionMethod, e.g. TurningFeatureA asserted with ProductionMethod of “forging”?’ can arise. In the present MRO, such an assertion is allowed. However, controlling this is easily possible through additional ICs and by formally defining the new concepts like *Turn- ingFeature* and *Forging* within the MRO. However, to maintain a core nature of the MRO, this level of formalisation has been left to the domain specialists. A domain ontology which commits to the MRO may introduce a set of ICs which would not allow wrong linkages, e.g. ForgingMethod cannot be linked to a TurningFeature. On the other hand, this points to the possible extension of the ontology in certain directions and this is discussed at the end of the paper.

Similar to the ICs for ProductionFeature, ICs have been defined throughout the MRO which capture the semantics of the most important concepts from all the eight categories of concepts. For example, when asserting the *Forms* of *Features*, the system asks the users to define the *Parameters* of *Forms* and when asserting *Family* and *PartFamily* facts, the system asks the user to define their criteria according to the formal definitions. The system also makes inferences, for example, as a result of the realisation of a *ProcessPlan*, a *RealisedPart* fact is inferred and a new *ProcessPlanVersion* is also asserted in the KB through automatic inference (A *ProcessPlan* is ‘a sequence of events which describes the procedure for the production of a *Part*' and a *ProcessPlanVersion* 'stores the historical information about a *ProcessPlan* e.g. different historical version of a *ProcessPlan*') All this is controlled through inference rules and ICs. This makes the ontology robust and provides a structure that can be used as basis for developing semantically sound application-specific ontologies.

7.3 Developing application-specific ontologies and interoperability

The *Feature* concepts from the MRO are used to show the ability of the MRO to support the development of semantically sound application-specific ontologies. As examples of application-specific ontologies, an AeroEngineDiscDesignOntology and an AeroEngineDiscProductionOntology are developed as shown in Figure 5. The concepts in these ontologies are identified from the industrial study. As shown, the two ontologies have different design and production feature concepts for the aero engine discs. These ontologies provide the structures to capture the details about actual disc features for design and production domains. The views showed in the bottom half of Figure 5 represent the instances of the application-specific ontologies that capture the facts about actual design and production features of a disc The ontological structure can be instantiated several times to capture facts about any number of actual design and production features on discs.

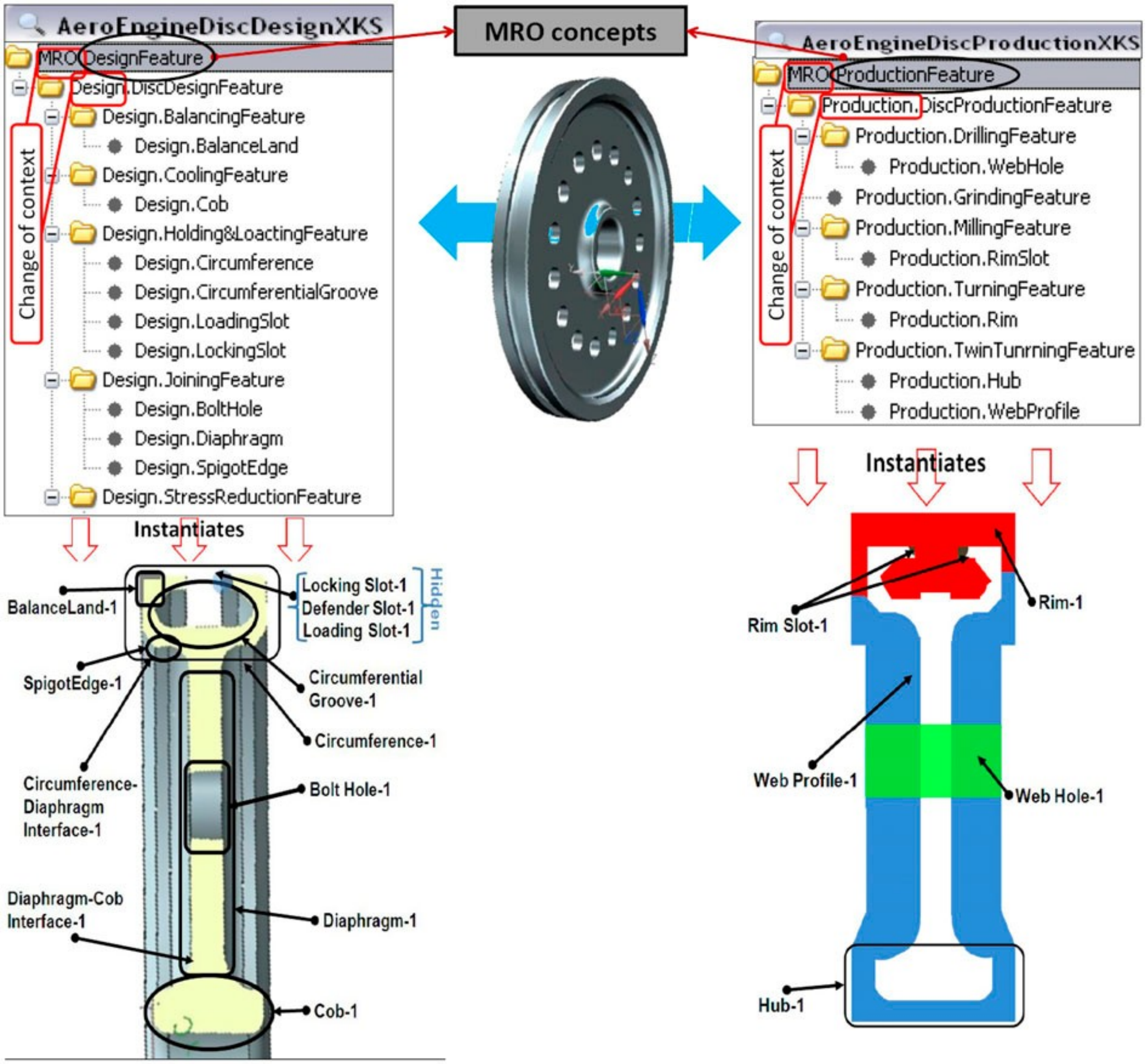


Figure 5. Application-specific ontologies developed from the MRO and the instantiated disc design and production views.

Table 2. Facts assertions for application-specific ontologies and their results.

Sr	Assertion	Result of Assertion			
			Result	ICs Violated	IC message
1	DiscDesignFeature	Without any attribute	✗	IC1: Hard IC IC2: Hard IC IC3: Soft IC IC4: Hard IC	IC1: An attribute of interest need to be defined for the asserted Feature IC2: A Form need to be defined for the asserted FormFeature IC3: An associated Part may be defined for the asserted PartFeature IC4: A Function need to be defined for the asserted DesignFeature
		With Function, Part and Form	☑	None	None
2	DiscProductionFeature	Without any attribute	✗	IC1: Hard IC IC2: Hard IC IC3: Soft IC IC4: Hard IC	IC1: An attribute of interest need to be defined for the asserted Feature IC2: A Form need to be defined for the asserted FormFeature IC3: An associated Part may be defined for the asserted PartFeature IC4: A ProductionMethod need to be defined for the asserted ProductionFeature
		With ProductionMethod, Part and Form	☑	None	None

The top portion of Figure 5 shows two core concepts from the MRO i.e. *DesignFeature* and *ProductionFeature* being used to develop the respective AeroEngineDiscDesignOntology and the AeroEngineDiscProductionOntology. The shift from the MRO to the application-specific ontologies is evident from the 'change of context' as show in Figure 5. The hierarchies of concepts of these ontologies are also shown in Figure 5. The bottom portion of Figure 5 shows that the application-specific ontologies are instantiated to capture the details about design and production features of the disc. The use of MRO concepts to develop the application-specific ontologies means that the application-specific ontologies agree on the MRO and that the semantics of MRO are inherited into them. While this example focuses on the use and extension of *Feature* concepts, all the other MRO concepts are also present at a basic layer for the application-specific ontologies. The verification of semantics consistency in the application-specific ontologies is done using the same method as explained in Section 7.1. The results are listed in the Table 2.

As shown in Table 2, the assertions for concepts from both application-specific ontologies, i.e. *DiscDesignFeature* and *DiscProductionFeature* were cancelled when no attributes were defined for the asserted facts. This happened due to the violation of a number of hard and soft ICs. All the ICs that have been violated belong to the MRO. Moreover, similar to the experiment in Section 7.1, when the same facts were asserted with their forms, parts, functions and production methods, they were successfully committed without any ICs being violated. The concepts *DiscDesignFeature* and *DiscProductionFeature* as shown in Table 2 belong to the application-specific ontologies from design and production domains. The experiment on fact assertion for these concepts leads to the conclusions that have been reported as follows:

- The MRO supports the development of application-specific ontologies.
- The application-specific product design and production ontologies inherit the semantics from the MRO, and thus possess semantic integrity and robustness.
- The application-specific product design and production ontologies can be developed within their own contexts while still being committed to the MRO.

8. Discussion and conclusions

8.1 Discussion

For the MRO to be effective on a broader scale in industry and in the research community, this should be used for applications beyond the narrow area within production domain addressed in this paper. The MRO should also be demonstrated for its successful use by other research and industrial projects. In this regard, it is pleasing to report that the MRO has successfully been used in two ‘Strategic Affordable Manufacturing in the UK with Leading Environmental Technology (SAMULET)’ projects, i.e. SAMULET-5.6.1 and SAMULET-3.7.3.

Elements of the MRO have been used and extended in SAMULET-5.6.1 to support the development of a manufacturing knowledge maintenance system which mainly used the UML models of *Feature* and *PartFamily* concepts. SAM- ULET-5.6.1 has defined methods for the capture and maintenance of manufacturing knowledge in the context of sharing across product design and production.

In SAMULET 3.7.3, MRO was used to contribute to the development of a life cycle knowledge desktop (LCKD), where an OWL-based version of MRO was developed and extended. LKCD offered the ability to access multiple sources of information by utilising, rather than recreating, existing knowledge. MRO particularly supported the development and delivery of a high-level manufacturing ontology for the LCKD.

The MRO was itself developed as part of the interoperable manufacturing knowledge system (IMKS) project to provide a reference library of computationally interpretable manufacturing concepts. A prototype PLM support system has been built as a result of the IMKS project which is based on the MRO.

The successful use of the MRO contributes to the establishment of the MRO concept as one that has the potential to act as reference manufacturing ontology for several product lifecycle disciplines.

The MRO needs further industrial validation, which requires further research work. For example, there can be an argument that the MRO is too generic for direct use by industry because concepts such as *TurningFeature* and *DrillingFeature,* which are not included in the MRO, could have been included, because such concepts are applicable to a wide range of manufacturing organisations. Similar argument applies to the extension of the MRO for different *ManufacturingProcess* concepts (e.g. *TurningProcess, MillingProcess* etc.), *ProductionMethod* concepts and so on.

However, a boundary line had to be drawn between reference manufacturing concepts and the application-specific concepts. Concepts such as *TurningFeature* and *MillingProcess* have not been included because the MRO was aimed at providing a basic structure with formal semantics that supports the development of semantically sound and consistent application-specific ontologies without enforcing the use of specific concepts such as *TurningFeature, MillingFeature*, etc. The more specific the reference ontology is, the more difficult it is for the application specific domains to utilise the concepts.

The MRO can however, still be extended to provide concepts that are directly more useful for industry. Each industry may have its own preference of concepts. Therefore, research should be conducted to extend the reference set of formally defined concepts for manufacturing processes, manufacturing resources, manufacturing facilities and realised parts and understand their relevance to a range of manufacturing industries. Similarly, research is required to define reference concepts for other areas within production, such as assembly, casting, rapid prototyping and forging. Similarly, reference concepts for other lifecycle domains such as services, operations and disposal should also be researched.

The application-specific ontologies developed are based on the examples from the aero engine industry. For the MRO to have broader application, it should be exploited for the development of application-specific ontologies in other manufacturing areas, e.g. ontologies for automotive industry applications.

The MRO can also be explored further for a more comprehensive set of design concepts. In this ontology, different design requirements, i.e. functional requirements, assembly requirements, stress requirements, etc. have all been merged into the design function. Although this is correct at a generic level, in practice, different specialists such as stress analysis, thermal analysis, assembly, and others will have their own more specific requirements that need to be understood.

For the MRO to be implemented as a software tool, i.e. a computational knowledge sharing system, the MRO should be explored further in the areas discussed above and a broader and comprehensive manufacturing ontology covering all the different relevant areas useful for the manufacturing industry should be formally defined. The ontology and the knowledge base system should be developed into a tool by developing a front end knowledge management interface. The knowledge management tool can also be developed by integrating the ontology and the KB into a product lifecycle management system such as Siemens PLM (2012).

The research work reported in this paper is a step towards forming a formal MRO and shows the potential to extend this approach to a much broader range of concepts to produce a full MRO.

8.2 Conclusions

The following set of conclusions has been drawn from the discussion.

- It has been shown that ontology (The MRO) of a comprehensive set of formally defined manufacturing concepts can eradicate the problem of inappropriate interpretation of concepts by providing a verifiable formal semantic base.
- It has been shown that through the adapted ontology development methodology, a semi-formal MRO can be developed and used as the base for formal definition.
- It has been shown that the semantics of concepts have been successfully captured in formal logic, which enabled the computer systems to interpret the semantics of concepts and respond in accordance with the formal
- definition of the concepts.
- It has been shown that MRO has the ability to support the development of semantically sound application-specific ontologies for product design and production.

Note

1. Attribute is not used to describe the objects or events and they are present in ontology on their own. The objects or events are not subsumed under the concept 'attribute' and there is no concept named 'attribute' in the MRO. However, *Features* are linked to their defining attributes which can be object(s) or event(s) through the relation *hasAttributeOfInterest*.

References

Abdul-Ghafour, S., G. Parisa, S. Behzad, and P. Eliane. 2007. "A Common Design Feature Ontology for Product Data Semantics Interoperability." In *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, edited by T. Y. Lin, Laura Haas, Janusz Kacprzyk, Rajeev Motwani, Andrei Broder, and Howard Ho, 443–446. Washington, DC: IEEE Computer Society.

Abdul-Ghafour, S., P. Ghodous, B. Shariat, E. Perna, D. D. Frey, S. Fukuda, and G. Rock. 2011. *Ontology Development for the Integration of CAD Models in a Collaborative Environment Improving Complex Systems Today* (207–214). London: Springer.

Anjum, N. A., J. A. Harding, R. I. Young, and K. Case. 2012. "Mediation of Foundation Ontology Based Knowledge Sources." *Computers in Industry* 63 (5): 433–442.

Blomqvist, E., and A. Öhgren. 2008. "Constructing an Enterprise Ontology for an Automotive Supplier." *Engineering Applications of Artificial Intelligence* 21 (3): 386–397.

Borgo, S., and P. Leitão. 2007. *Foundations for a Core Ontology of Manufacturing*. Portugal: Bragança.

Brunnermeier, S. B., and S. A. Martin. 1999. *Interoperability Cost Analysis of the U.S. Automotive Supply Chain*. Gaithersburg, MD: National Institute of Standards and Technology.

Bussler, C., V. Tannen, I. Dulaki, P. Tan, S. Madnick, and K. Tan. 2005. *Context Mediation in the Semantic Web: Handling OWL Ontology and Data Disparity Through Context Interchange. Semantic Web and Databases*. Berlin: Springer.

Chang, X., A. Sahin, and J. Terpenny. 2008. "An Ontology-based Support for Product Conceptual Design." *Robotics and Computer-Integrated Manufacturing* 24 (6): 755–762.

Chang, X., R. Rai, and J. Terpenny. 2010. "Development and Utilization of Ontologies in Design for Manufacturing." *Journal of Mechanical Design* 132 (2): 021009.

Chen, D., G. Doumeingts, and F. Vernadat. 2008. "Architectures for Enterprise Integration and Interoperability: Past, Present and Future." *Computers in Industry* 59: 647–659.

Chen, Y.-J., Y.-M. Chen, and H.-C. Chu. 2009. "Development of a Mechanism for Ontology-based Product Lifecycle Knowledge Integration." *Expert Systems with Applications* 36 (2): 2759–2779.

Chen, Y., and C. Wei. 1997. "Computer-Aided Feature-Based Design for Net Shape Manufacturing." *Computer Integrated Manufacturing Systems.* 10: 147–164.

Chungoora, N. 2010. "A Framework to Support Semantic Interoperability in Product Design and Manufacture." PhD thesis, Wolfson School of Mechanical & Manufacturing Engineering, Loughborough University, Loughborough.

Chungoora, N., and R. I. M. Young. 2010. "The Configuration of Design and Manufacture Knowledge Models from a Heavyweight Ontological Foundation." *International Journal of Production Research* 49 (15): 4701–4725.

Chungoora, N., O. Canciglieri, et al. 2010. "Towards Expressive Ontology-based Approaches to Manufacturing Knowledge Representation and Sharing." *Taylor & Francis* 23: 1059–1070.

CIMOSA-Association. 1996. "CIMOSA – Open System Architecture for CIM." Technical Baseline, Version 3.2.

Corcho, O., M. Fernandez-Lopez, and A. Gomez-Perez. 2003. "Methodologies, Tools and Languages for Building Ontologies: Where is their Meeting Point." *Elsevier Science Publishers BV* 46: 41–64.

Dartigues, C., P. Ghodous, M. Gruninger, D. Pallez, and R. Sriram. 2007. "CAD/CAPP Integration Using Feature Ontology." *Concurrent Engineering* 15 (2): 237–249.

Deshayes, l., S. Foufou, and M. Gruninger. 2007. “An Ontology Architecture for Standards Integration and Conformance in Manufacturing.” In *Advances in Integrated Design and Conformance in Manufacturing II*, edited by S. Tichkiewitch, M. Tollenaere, and P. Ray, 32–341. Amsterdam: Springer.

Elkin, C., and R. Greiner. 1993. “Building Large Knowledge-Based Systems: Representation and Inference in the Cyc Project: D. B. Lenat and R. V. Guha.” *Artificial Intelligence* 61: 41–52.

ESPRIT-Consortium-AMICE. 1993. “CIMOSA – Open Systems Architecture for CIM, 2nd revised and extended edition Research Report”, ESPRIT Project 688/5288. Berlin: Springer-Verlag.

Feng, S. C., and E. Y. Song. 2003. “A Manufacturing Process Information Model for Design and Process Planning Integration.” *Journal of Manufacturing Systems* 22: 1–15.

Fenves, S. J., S. Foufou, C. Bock, and R. D. Sriram. 2008. “CPM: A Core Model for Product Data.” *Journal of Computing and Information Science in Engineering* 8 (1): 014501-1–014501-6.

Fernández-López, M. 1999. “Overview of Methodologies for Building Ontologies.” *IJCAI99 Workshop on Ontologies and Problem-Solving Methods: Lessons Learned and Future Trends*. Stockholm.

Fox, M. S., and M. Gruninger .1997. “On Ontologies And Enterprise Modelling.” In *International Conference on Enterprise Integration Modelling Technology*, edited by Kurt Kosanke, and James G. Nell. Torino: Springer.

Gallaher, M. P., A. C. O’Connor, et al. 2009. *Cost Analysis of Inadequate Interoperability in the US Capital Facilities Industry*. Gaithersburg, MA: US Department of Commerce Technology Administration, National Institute of Standards and Technology (NIST).

Gangemi, A., and S. Borgo 2004. “Core Ontologies in Ontology Engineering 2004 (Un)Successful Cases and Best Practices for Ontology Engineering: Reusing Well-founded Ontologies for Domain Content Specification.” *Proceedings of the EKAW’04 Workshop on Core Ontologies in Ontology, Engineering*. Northamptonshire, UK.

Gruninger, M., and M. Fox. 1995. “Methodology for the Design and Evaluation of Ontologies.” *IJCAI'95, Workshop on Basic Ontological Issues in Knowledge Sharing*, Montreal, Canada, April 13, 1995.

Gunendran, A. G., and R. I. M. Young. 2010. “Methods for the Capture of Manufacture Best Practice in Product Lifecycle Management.” *International Journal of Production Research* 48 (20): 5885–5904.

Hakimpour, F. 2003. “Using Ontologies to Resolve Semantic Heterogeneity for Integrating Spatial Database Schemata.” PhD thesis, University of Zurich, Zurich.

Harding, J. A., and B. Yu. 1999. “Information-centred Enterprise Design Supported by a Factory Data Model and Data Warehousing.” *Elsevier Science Publishers B. V.* 40: 23–36.

Harding, J. A., K. Popplewell, and D. Cook. 2003. “Manufacturing System Engineering Moderator: An Aid for Multidiscipline Project Teams.” *International Journal of Production Research* 41(9): 1973–1986.

Hoque, A. S. M., and T. Szecsi. 2008. “Designing using Manufacturing Feature Library.” *Journal of Materials Processing Technology* 201: 204–208.

IEEE Std. 1999. *610–1991: IEEE Standard Computer Dictionary. A Compilation of IEEE Standard Computer Glossaries*. Computer Society.

Iode, H. 2010. *Ontology Libraray Refernce*. Baltimore, MD: Highfleet's IODE/XKS Documentation.

ISO/IEC-TR-10000-1.1998. *Information technology – Framework and taxonomy of International Standardized Profiles – Part 3: Principles and Taxonomy for Open System Environment Profiles*. Geneva: International Organization for Standardization.

ISO/IEC-24707. 2007. *International Standard, First edition 2007–10-01 Information Technology – Common Logic (CL): A Framework for a Family of Logic Based Languages*. Geneva: International Organization for Standardization.

ISO/TS-10303-1022. 2004. *Industrial Automation Systems and Integration– Product Data Representation and Exchange– Part 1022: Application Module: Part and Version Identification*. Geneva: International Organization for Standardization.

ISO-10303-224. 2006. *Industrial Automation Systems and Integration – Product Data Representation and Exchange – Part 224: Application Protocol: Mechanical Product Definition for Process Planning Using Machining Features*. Geneva: International Organization for Standardization.

ISO-10303-239. 2006. *Industrial Automation Systems and Integration – Product Data Representation and Exchange – Part 239: Application Protocol: Product Life-Cycle Support*. Geneva: International Organization for Standardization.

ISO-13584. 2001. *Industrial Automation Systems and Integration – Parts Library* . Geneva: International Organization for Standardization.

ISO-15531-1. 2004. *Industrial Automation Systems and Integration – Industrial Manufacturing Management Data – General Overview: Part 1*. Geneva: International Organization for Standardization.

Lagos, N. and R. M. Setchi. 2007. “A Manufacturing Ontology for E-Learning.” In *Proceedings of the 3rd Intelligent Production Machines and Systems Conference*, edited by D. T. Pham, Eldaw E. Eldukhri, and Anthony J. Soroka, 552–557. Cardiff: PROMS Network of Excellence.

Lee, J.-H., and H.-W. Suh. 2008. “Ontology-based Multi-layered Knowledge Framework for Product Lifecycle Management.” *Concurrent Engineering* 16 (4): 301–311.

Leimagnan, S., A. Siadat, J. Dantan, and A. Semenenko. 2006. “MASON: A Proposal for an Ontology for Manufacturing Domain.” *Proceedings of the IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS’06), IEEE Xplore*.

Lin, H. K., and J. A. Harding. 2007. "A Manufacturing System Engineering Ontology Model on the Semantic Web for Inter-Enterprise Collaboration." *Computers in Industry* 58 (5): 428–437.

Masolo, C., S. Borgo, et al. 2003. *Ontology Library*, *WonderWeb Project*. Italy: Laboratory for Applied Ontology - ISTC-CNR.

Molina, A., and R. Bell. 1999. "A Manufacturing Model Representation of a Flexible Manufacturing Facility." *Journal of Engineering Manufacture: Proceedings of the Institution of Mechanical Engineers* 213 (Part B): 225–246.

Nagel, R. N., W. W. Braithwaite, and P. R. Kennicott. 1980. "Initial Graphics Exchange Specification IGES, version 1.0."

Noy, N. F., and D. L. McGuinness. 2000. *Ontology Development 101: A Guide to Creating Your First Ontology*. Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880.

Patil, L., D. Dutta, and R. Sriram. 2005. "Ontology-based Exchange of Product Data Semantics." *IEEE Transactions on Automation Science and Engineering* 2 (3): 213–225.

Sarder, M. D. B. 2006. "The Development of Design Ontology for Products and Processes." PhD Thesis, The University of Texas, Arlington, TX.

Schlenoff, C., M. Gruninger, F. Tissot, J. Valois, J. Lubell, and J. Lee. 2000. "ISO-18629 The Process Specification Language (PSL): Overview and Version 1.0 Specification, NISTIR 6459." Gaithersburg, MD: National Institute of Standards and Technology.

Semere, D., T. S. Dislshad, and L. Bengt. 2007. *Machining ontology and knowledge modelling*. Sweden: Swedish Production Symposium.

Siemens-website. 2012. http://www.plm.automation.siemens.com/en_us/products/teamcenter/.

Simpson, J. A., R. J. Hocken, and J. S. Albus. 1982. "The Automated Manufacturing Research Facility of the National Bureau of Standards." *Journal of Manufacturing Systems* 1: 17–32.

Staab, S., R. Studer, et al. 2001. "Knowledge Processes and Ontologies." *Intelligent Systems, IEEE* 16 (1): 26–34.

Swartout, B., P. Ramesh, L. Knight and T. Russ. 1997. *Toward Distributed Use of Large-Scale Ontologies*. AAAI Symposium on Ontological Engineering, Standford, CA.

Terzi, S., H. Panetto, M. Gérard Morel, and M. Garetti. 2007. "A Holonic Metamodel for Product Traceabily in PLM." *International Journal of product lifecycle management* 2 (3): 253–289.

Todd, R. H. 1994. *Fundamental Principles of Manufacturing Processes* (1). New York: I. P. Inc, Industrial Press.

True, M., and C. Izzi. 2002. "Collaborative Product Commerce: Creating Value Across the Enterprise." *Ascet* 4: 27–35.

Tursi, A., H. Panetto, et al. 2009. "Ontological Approach for Products-Centric Information System Interoperability in Networked Manufacturing Enterprises." *Annual Reviews in Control* 33 (2): 238–245.

Uschold, M., and M. King. 1995. "Towards a Methodology for Building Ontologies." In *IJCAI95 Workshop on Basic Ontological Issues in Knowledge Sharing*, edited by Chris S. Mellish. Montreal: M. Kaufmann.

Uschold, M., M. King, S. Moralee, and Y. Zorgios. 1998. "The Enterprise Ontology" *The Knowledge Engineering Review* 13: 31–89.

Usman, Z., R. I. M. Young, N. Chungoora, C. Palmer, K. Case, and J. A. Harding. 2011. *A Manufacturing Core Concepts Ontology for Product Life Cycle Interoperability International Ifip Working Conference on Enterprise Interoperability*. Stockholm, Sweden: Springer.

Vichare, P., A. Nassehi, S. Kumar, and S. T. Newman. 2009. "A Unified Manufacturing Resource Model for representing CNC Machining Systems." *Robotics and Computer-Integrated Manufacturing* 25: 999–1007.

Young, R. I. M., A. Gunendran, A. F. Cutting-Decelle, and M. Gruninger. 2007. "Manufacturing Knowledge Sharing in PLM: A Progression Towards the Use of Heavy Weight Ontologies." *International Journal of Production Research* 45 (7): 1505–1519.

Zhao, J., W. M. Cheung, et al. 1999. "A Consistent Manufacturing Data Model to Support Virtual Enterprises." *International Journal of Agile Management Systems* 1 (3): 150–158.