Semantic Reconciliation across Design and Manufacture Knowledge Models: a Logic-Based Approach

CHUNGOORA, N.¹, YOUNG, R.I.M.¹

¹Wolfson School of Mechanical and Manufacturing Engineering, Loughborough University, Loughborough, Leicestershire, LE11 3TU, UK.

Corresponding Author: Mr. Nitishal Chungoora e-mail: <u>N.Chungoora@lboro.ac.uk</u>

Abstract

Ontology-based models of product design and manufacture are becoming increasingly important in the effort towards achieving interoperability among various stakeholders within and across product lifecycle systems. However, in the eventuality of having to interoperate between multiple ontology-based models, with the intention of sharing semantic knowledge among them, the process still remains a difficult one. Although the concept of ontology mapping/matching has been developed as a means to interoperate across ontology-based models, yet the concept has remained relatively weak in terms of its ability to enable the formalisation and verification of cross-model semantic correspondences. In this paper, improved concepts to achieve semantic reconciliation are being investigated in the context of the Semantic Manufacturing Interoperability Framework (SMIF). The approach uses a Common Logic-based underpinning for enabling the evaluation and verification of cross-model correspondences. The approach has been successfully tested by applying the relevant logic-based mechanisms in order to show the reconciliation of two individually developed machining hole feature knowledge models. Through this, it has been demonstrated that the approach enables semantic reconciliation of important structures within ontology-based models of design and manufacture.

Keywords: Design and Manufacture, Heavyweight Ontology, Common Logic, Ontology Mapping, Semantic Alignment, Interoperability.

1 INTRODUCTION

Nowadays, ontology engineering is being exploited by different parties involved in collaborative product development in order to formalise models of knowledge, while fostering a basis for sharing meaning (i.e. semantics) in computational form. Ontology-based approaches are considered an important facet of industrial knowledge management strategies and the deployment of best practice Knowledge Bases (KBs) [22]. These help support the production of better, faster and cheaper products, through the reuse of formal knowledge that drives key engineering decisions during collaborative product development.

In Product Lifecycle Management (PLM), knowledge may be formalised for the different nodes within the product lifecycle such as conceptual design, detailed design, manufacturing and operation [1]. This results in multiple viewpoints associated to the representation of artifacts [2, 3] leading to different representations as well as alternative representations of similar concepts. Consequently, the problem of semantic heterogeneity emerges, the latter causing the occurrence of semantic mismatches across ontology-based models. Semantic mismatches are related to the incompatibilities of data and information structures [5, 6] as well as interoperability problems related to manufacturing applications [7], which need to be resolved in the effort towards the timely exchange of design and manufacturing ontology-based knowledge.

A considerable amount of effort has been input from the ontology mapping/matching community so as to address the obstacles to ontology interoperability brought about by semantic mismatches. However, there still exist a number of challenges to be overcome in order to progress towards improved ontology mapping/matching techniques [8]. It is also evident, from the experiments based on current methods, that ontology mapping/matching has not been given due attention in product design and manufacture, primarily since the latter remains an expert field of knowledge with very specific content and issues [9]. This paper hence identifies a Semantic Manufacturing Interoperability Framework (SMIF) [9, 10], which supports the exchange of semantics across ontology-based models of product design and manufacture.

The approach uses the semantic capability offered by a manufacturing foundation ontology from which ontology-based models can be specialised. In addition to this, the expressive Common Logic-based underpinning used in the framework provides sufficient coverage for the development of logic-based semantic reconciliation mechanisms. The latter facilitate automated reconciliation across pairs of specialised models that have been based on the same foundation. During the implementation of the framework, two ontology-based models of hole feature machining have been individually developed using the manufacturing foundation ontology, and the relevant semantic reconciliation methods have been deployed to successfully reconcile across the two models. We, therefore, stipulate that the approach contributes to (1) an understanding of new and expressive ontological formalisms like Common Logic [26] applied to ontology mapping/matching in design and manufacture and (2) improved ways of specifying rigorous and reusable reconciliation mechanisms for ontology-based knowledge sharing.

2 SEMANTIC RECONCILIATION ACROSS ONTOLOGY-BASED MODELS

2.1 Techniques for Ontology Mapping

The area of ontology mapping/matching has been a key direction to tackle semantic heterogeneity issues, through the reconciliation of the semantics carried by ontology-based models. Several overlapping perspectives on the types of ontology mapping/matching methods have been suggested [11-14]. There is almost general consensus over the types of methods that can be applied in ontology mapping/matching. Figure 1, identifies and summarises these methods.



Figure 1: Common methods used for ontology mapping/matching

Ontology mapping/matching methods include (1) techniques that focus on combining (i.e. merging) two ontologies to construct a new ontology from the individual ontologies, (2) tools that compile a transformation function that transforms a given ontology into another based on the transformation rules specified [12], (3) methods that concentrate on establishing a collection of binary relations between the vocabularies of two ontologies (i.e. alignment) [11] and (4) methodologies that enable specific portions of two ontologies to be reconciled, through the definition of mappings via an intermediate articulation ontology. It is to be noted that although some of these methods fit very well into these four categories, others may occur as hybrids of the common ontology mapping/matching methods identified in Figure 1.

A large number of ontology mapping/matching methods have been investigated such as in [12, 15, 16]. Current methods indicate that there is presently a spectrum of these that rely on lexical similarity matching, which from a semantic interoperability viewpoint is not optimal. This is because in product design engineering, similar terms are often being employed across different groups to refer to disparate concepts. Hence, it is only via the semantics associated to these terms that existing differences can be identified, thereby implying the need to expressively capture semantics in the first place.

On the other hand, it has also been shown that ontological formalisms like the Web Ontology Language (OWL) support a number of built-ins for ontology mapping [23, 24]. However, the semantics of these built-ins are not expressive enough to map across the semantic content of manufacturing ontologies and their associated KBs. Furthermore, there exist ongoing requirements [8-10] to support the development of mapping/matching techniques that (1) can be formally interpreted and verified, (2) are focused at the identification and potential resolution of semantic mismatches, (3) can enable reconciliation at several structural levels of ontology-based models including the instance level and (4) perform better in terms of accuracy, automation and the duration taken to resolve cross-model correspondences.

2.2 Concepts for Semantic Reconciliation

Figure 2 illustrates the primary concepts for semantic reconciliation, under investigation in this paper. A Common Logic-based application (A) is used to deploy the concepts for semantic reconciliation, predominantly because Common Logic allows the representation of highly expressive semantics over which complex reasoning can be performed. The Foundation Layer (B) holds the essential ontological structures of a manufacturing foundation ontology. These ontological structures can be configured to address the semantics of specialised ontology-based models (C). For a given pair of ontology-based models that have both been configured from the Foundation Layer, the manufacturing foundation ontology firstly enables specific portions of the specialised ontology-based models to be articulated (D), based on the lineage of specialised concepts.



Figure 2: Overview of concepts for semantic reconciliation

To impart further semantic reconciliation capability, reusable mapping models (E) can be defined using Common Logic. This helps to improve the formality of alignment relations that are held in mapping models. The expressiveness of the Foundation Layer also facilitates reconciliation at various levels of granularity across specialised ontology-based models. Furthermore, the approach is supported by state-of-the-art querying mechanisms (F), which provide high performance levels during the processing of queries for identifying alignments across specialised ontology-based content.

3 FRAMEWORK TO SUPPORT SEMANTIC INTEROPERABILITY

The Semantic Manufacturing Interoperability Framework (SMIF) [9, 10] has been developed in order to support the ability to formally reconcile the semantics of ontology-based models of product design and manufacture, where these models have

been based on a shared foundation ontology. Figure 3 depicts the framework and its four constituent layers, which are elaborated in the next sub-sections.



Figure 3: Semantic Manufacturing Interoperability Framework (SMIF)

In this paper, emphasis is laid on the upper two layers of the framework, where the concepts for semantic reconciliation identified in section 2.2 have been applied. It is to be noted that to enable the exploration of the breadth of concepts used in the framework, the scope of this work has been placed on simple product ontologies and KBs capturing feature descriptions involving hole features in design and manufacture. The implications of each layer of the framework have been pursued based on the chosen scope.

3.1 Foundation Layer

The Foundation Layer provides the essential basis for sharing meaning and comprises two characteristic elements namely (1) a Common Logic-based ontological formalism over which (2) a heavyweight manufacturing ontological foundation (manufacturing foundation ontology) is constructed. This Common Logic-based formalism, known as the Knowledge Frame Language (KFL) has been developed by Highfleet Inc. (formerly Ontology Works Inc.) [17]. The formalism holds a rigorously-defined meta-model which imparts the necessary syntax and expressive semantics, which supports the "heavyweight" nature of the foundation. The benefit of the application of Common Logic lies in the latter being a First Order Logic language for knowledge interchange that provides a core semantic framework for logic together with the basis for a set of syntactic forms (dialects) all sharing common semantics [18].

The manufacturing foundation ontology formalises core entity information and process semantics. To capture entity information, for the meaningful description of reusable feature-based semantics in product representation, the fundamentals from the revised Core Product Model (CPM) [19] and those from ISO 10303 AP224 [20] have been adapted and formalised to the framework needs. The accommodation of process semantics involves the inclusion of concepts from the relevant ontological theories present in the Process Specification Language (PSL) [21]. Figure 4 identifies examples of semantic structures within the manufacturing foundation ontology. Some of the classes present in the taxonomy for the manufacturing foundation ontology are shown. The figure highlights the class "Round_Hole" (G) and its natural language description (H), together with the relations (I) that apply to the class "Feature" and hence "Round_Hole".



Figure 4: Example of classes, relations and axioms in the Foundation Layer

These relations include, for example, the relation "compound" (J) for the statement of complex features that are composed of singleton features, and the relation "holds_feature" (K), which is used to associate features to artifacts. Figure 4 also

depicts two of the Integrity Constraints (ICs) (L) relevant to the class "Round_Hole". These ICs, expressed as logic-based axioms, capture the semantic prescriptions for the interpretation of "Round_Hole" instances based on relevant geometric parameters, their relationships and constraints (M). A similar understanding also applies to the formal definition of all foundation level concepts present in the taxonomy in Figure 4.

3.2 Domain Ontology Layer

At this level of the framework, semantic structures from the Foundation Layer can be reused and specialised for the construction of knowledge models (i.e. customised ontology-based models with their associated KBs). A number of ontological mechanisms are supported in order to allow knowledge models to be integrally defined. Such mechanisms include, for example, (1) identifiers for knowledge models which work similar to namespaces, (2) ontological relationships that allow the specialisation of the foundation taxonomy, (3) the inclusion of domain-defined ICs as long as these do not violate foundation ICs and (4) the representation of discrete domain knowledge through instantiation.

The advantage of exploiting a shared foundation ontology for the configuration of knowledge models is related to enabling sufficient flexibility during specialisation, while ensuring that the configured models respect the core established foundation semantics. This helps to reduce, if not eliminate, obvious types of semantic mismatches such as relation mismatches and class aggregation level mismatches [4, 9] to occur between individually-developed knowledge models.

3.3 Semantic Reconciliation Layer

The Semantic Reconciliation Layer supports the mechanisms necessary for reconciling pairs of knowledge models, with the intention of facilitating knowledge sharing. To achieve reconciliation requires following an ordered ontology mapping process, in which a logic-based semantic alignment method enables knowledge model content to be matched through the use of mapping models. The Semantic Reconciliation Layer thus combines the semantic matching techniques of ontology alignment and articulation (see section 2) via a shared semantically-defined foundation ontology, a method that has so far not been exploited [13].

3.3.1 Ontology Mapping Process Concepts

Figure 5 conceptually shows the steps involved in the ontology mapping process for a pair of knowledge models to be reconciled. The diagram identifies a first stage where the identifiers of both models are adjusted (N). In this stage, the identifiers of the two models are renamed to the pre-defined identifiers called "DomainX" and "DomainY", one for each model respectively. The two knowledge models are then imported under a single Object Management System (0). The latter stage brings the ontology-based content from the two models under a single KB, while keeping the ontology-based content still distinct to the initial models.



Figure 5: Ontology mapping process concepts in the Semantic Reconciliation Layer

After the importing stage is performed, the alignment stage is initiated (P). The alignment stage requires loading the appropriate mapping model in the single Object Management System. Mapping models contain "semantic mapping concepts". These semantic mapping concepts are written in KFL, and serve as formally-defined and reusable built-in relations for matching cross-model content. On one hand, the consequence of using a manufacturing foundation ontology allows for the lineage of domain-specialised semantics to be traced, which conveys basic reconciliation via articulation. The combination of both articulation and alignment (Q) provides enhanced semantic reconciliation. It is to be noted that while the development of mapping models requires significant human input, the alignment process is essentially fully automatic as it is logic-based. Moreover, mapping models are reusable for any pair of knowledge models that require reconciliation.

3.3.2 Semantic Mapping Concepts in Mapping Models

Figure 6 illustrates a sample of two semantic mapping concepts targeted towards the reconciliation of domain-level classes and instances. It is seen that a semantic mapping concept consists of relations such as "classMappingRelation_018" (R) and "instanceMappingRelation_042" (S). In the case of "classMappingRelation_018" its formal definition is provided by logic-based conditions (T), informally captured in the reconciliation scenario (U). These conditions state that the semantic mapping concept "classMappingRelation_018" be inferred true between any two arguments ?x and ?y, provided both ?x and ?y are sub-classes of the foundation class "Round_Hole", where ?x and ?y are located in the "DomainX" and "DomainY" identifiers respectively.

It is important to notice in the logical conditions (T) the way in which the semantics of the Foundation Layer are used in the logic-based statements. For example, the meta-model relation "RootCtx.sup" helps to express taxonomical relationships while the meta-model relation "RootCtx.withinContext" expresses the identifier for an ontology-based argument. Notice also the appearance of the identifiers "DomainX" and

"DomainY" in the captured reconciliation scenario (U). This justifies the need for renaming knowledge model identifiers, discussed previously in section 3.3.1.



Figure 6: Understanding semantic mapping concepts in mapping models

In addition to formal semantics, a semantic mapping concept also includes informal remarks in the form of natural language for human interpretation. This is because alignments produced by matching systems may not be intuitively obvious to human users and, therefore, need to be explained [8]. These remarks generally include the informal way of interpreting a semantic mapping concept in a mapping model. In certain situations, depending on the reliability of a semantic mapping concept, other remarks may be added to reflect the uncertainties in a semantic mapping concept, or to flag unavoidable semantic mismatches even in the presence of alignments. In the case of the semantic mapping concept "classMappingRelation_018", the informal remarks required for human interpretation include the comments shown (V). Some of these comments relate to the behaviour of the semantics of "Round_Hole" controlled by the manufacturing foundation ontology. Other comments (W) relate to possible semantic mismatches that may still occur even after articulation and alignment.

The semantic mapping concept "instanceMappingRelation_042" (S) provides suitable reconciliation at the instance level of knowledge models to aid the interoperation of discretely-represented knowledge. The relation "instanceMappingRelation_042" is intended to establish an alignment between two cross-domain arguments *?holex* and *?holey* in the reconciliation scenario (X) provided (1) both are instances of "Round_Hole" located in the "DomainX" and "DomainY" identifiers respectively and (2) both share the commonality of having "through" hole bottom conditions. The logical conditions (Y) formally capture this reconciliation scenario. The accompanying remarks (Z) identify the informal comments for this semantic mapping concept.

3.4 Interoperability Evaluation Layer

After the ontology mapping process concepts are executed in the Semantic Reconciliation Layer, the approach requires running suitable queries for identifying established alignments across the content of knowledge models. In the framework,

queries are written in KFL form and are supported by appropriate query tools. An example of such a query is expressed in the KFL statement below:

(and (RootCtx.BinaryRel ?rel) (RootCtx.withinContext ?rel ModelA) (RootCtx.holdsArg ?rel 1 DomainX.EntityX) (RootCtx.holdsArg ?rel 1 DomainY.EntityY)

This query enables the retrieval of all semantic mapping concepts, coming from the mapping model with identifier "ModelA", that exist between the arguments "EntityX" and "EntityY", which are held in the "DomainX" and "DomainY" identifiers respectively. In the current approach, it has been recognised that the writing of such queries demands a good command of KFL. However, it has been shown that purposely-designed Graphical User Interfaces can be developed in order to semi-automatically generate complex queries without requiring prior knowledge of KFL [9].

4 IMPLEMENTATION AND RECONCILIATION OF DOMAIN MODELS

This section highlights an experimental implementation which tests the approach described so far. For this purpose, the Integrated Ontology Development Environment (IODE) developed by Highfleet Inc. [17] has been exploited because the latter handles semantic structures expressed in KFL. The proposed concepts for logic-based semantic reconciliation have been deployed in order to illustrate the extent of knowledge sharing across the ontology-based content coming from a pair of knowledge models. These models have been specialised from the manufacturing foundation ontology previously identified in Figure 4. Furthermore, the scope of the knowledge models involves simple product representations which hold several types of hole features defined from a machining perspective. This section also demonstrates the use of one mapping model for enabling the reconciliation of the Geometric Dimensioning and Tolerancing (GD&T) parameters carried by hole features.

4.1 Machining Hole Feature Ontology A

This knowledge model partly captured in Figure 7 has

"machiningHoleFeatureOntologyA" as identifier. The figure illustrates the specialisation of concepts from the Foundation Layer into domain-specific concepts. The taxonomy of entity information classes (A') is shown, where concepts such as "Reamed_Hole" (B') are configured from the existing foundation structures. In this case, the domain class "Reamed_Hole" appears as a sub-class of the foundation class "Round_Hole". Specific sub-classes of "Length_Measure", namely "Reamed_Hole_Depth" (C') and "Reamed_Hole_Diameter" (D') define specific dimensional parameters carried by "Reamed_Hole".

Domain-defined process concepts such as "Centre_Drilling" (E'), "Drilling" (F') and "Reaming" (G') are also present for describing executions of "Reamed_Hole_Making" (H') to obtain the process sequence for realising instances of "Reamed_Hole". Domain-specific semantic prescriptions are embedded by adding domain-defined ICs which adopt a similar form to the ones previously exposed in Figure 4. Concepts from the domain ontology have then been instantiated in order to represent discrete domain knowledge. One such instantiation is diagrammatically shown for the instance



"Reamed_Hole_A" (I') of the class "Reamed_Hole". This specific instance carries distinct shape aspects and transition features which hold concrete dimensional values.

Figure 7: Sample of ontology-based content in "Machining Hole Feature Ontology A"

4.2 Machining Hole Feature Ontology B

Another knowledge model using the "machiningHoleFeatureOntologyB" identifier has been developed from the Foundation Layer. Figure 8 depicts the taxonomy of process and entity information classes for the "Machining Hole Feature Ontology B". The class "Bored_Hole" (J') has been highlighted as well as selected sub-classes of the foundation class "Activity", namely "Rough_Boring" (K'), "Finish_Boring" (L') and "Bore_Hole_Making" (M'). These machining process concepts have been exploited to represent the process sequence that takes place during "Bore_Hole_Making" processes which produce instances of "Bored_Hole" such as "Bored_Hole_E" (N'). The latter is an instance which carries specific geometric and dimensional semantics.



Figure 8: Sample of ontology-based content in "Machining Hole Feature Ontology B"

4.3 Semantic Reconciliation

The ontology mapping process concepts previously explained in Figure 5 are applied to the pair of knowledge models. Firstly, the identifiers for "Machining Hole Feature Ontology A" and "Machining Hole Feature Ontology B" are renamed to "DomainX" and "DomainY" respectively, as shown in Figure 9 (O'). The adjustment of the identifiers for knowledge models is semi-automatic as it requires human intervention. Both models are then imported onto the same Object Management System (P'). During this stage, the specialised content from the models become articulated via the manufacturing foundation ontology, which acts as a general mediator for the two models.

A sample of the resultant taxonomy of the "Round_Hole" class, after the importing stage is completed, can be identified in Figure 9 (Q'). It is important to notice the presence of all domain-defined sub-classes of "Round_Hole", which have stayed distinct to their respective ontologies, for example, "Reamed_Hole" in "DomainX" and "Bored_Hole" in "DomainY". To complete semantic reconciliation, the required mapping model for enabling GD&T comparisons is loaded (R'), as part of the alignment stage. This enables semantic mapping concepts present in the mapping model

to be fed to the imported knowledge models. The alignment stage is fully automatic as it uses the logic-based underpinning described in section 3.3.2.

Adjust ide	ntifiers				
Replace					
Find Repla	ce Find in Files				
Find what :	machiningHoleFeatureOntologyA	(O')			
Replace with :	DomainX	Replace All	Im	port domain models	
Find what :	machiningHoleFeatureOntologyB			port domain models	
Replace with :	DomainY 💟	Report Log		Next	
		Loading Machining Hole Feature Ontology A.kfl Loading Machining Hole Feature Ontology B.kfl Load accepted Saving to the OMS Finished save. (P')		Back Cancel Help Load automatically	Foundation.Round_Hole DomainY.Bored_Hole DomainX.Centre_Drilled_Hole DomainX.Counterbore DomainX.Drilled_Hole DomainX.Drilled_Hole
		(F) Save automatically		Save automatically	DomainX.Reamed Hole
		Load mapping model	et	Neut	
		Report Log		Next	
		Loading GD&TModel.kfl Load accepted		Cancel	
		Finished save.	(R')	<u>Load automatically</u> Save automatically	
		100%			

Figure 9: Executing ontology mapping process concepts

4.4 Mapping Discovery

Consider Figure 10, which exemplifies the approach towards the discovery of mappings. Knowing two cross-domain arguments such as pairs of classes obtained from browsing the imported knowledge models, for example "Drilled_Hole" and "Bored_Hole", it is possible to formulate a query for finding all the correspondences that hold between these two classes. The structure of the query, written in KFL, is shown (S'). When this query is processed using the query tool in IODE, one semantic mapping concept is retrieved between the two classes "Drilled_Hole" and "Bored_Hole". The "classMappingRelation_018" (T') supports the knowledge behind the articulation and alignment of the two classes. This semantic mapping concept can be browsed as shown in Figure 11 (U'), so that the user is able to understand the mapping between the classes "Drilled_Hole".

In a similar fashion, cross-domain instances can also be evaluated for correspondences. In Figure 10, it is intended to retrieve all semantic mapping concepts that hold between "Reamed_Hole_A" and "Bored_Hole_E". The structure of the query has been portrayed (V'). When this query is executed using the query tool in IODE, six distinct semantic mapping concepts are retrieved. These semantic mapping concepts are browsed so that the user interprets the mappings that hold between the two instances. In this case, the following results apply:

- "instanceMappingRelation_023" (W') => Figure 11 (C'')
- "instanceMappingRelation_042" (X') => Figure 11 (D'')

- "instanceMappingRelation 048" (Y') => Figure 11 (E'')
- "instanceMappingRelation_054" (Z') => Figure 11 (F'')
- "instanceMappingRelation 059" (A") => Figure 11 (G")
- "instanceMappingRelation 068" (**B**") => Figure 11 (**H**")



Figure 10: Query structures and results



Figure 11: Browsing established semantic mapping concepts

4 DISCUSSIONS

The approach documented in this paper has demonstrated the constructive use of the latest developments in semantic-based technologies for enabling semantic reconciliation of product design and manufacture knowledge models. The capability to support a logic-based approach to semantic reconciliation derives from (1) the framework's ability to formally enable the definition of mapping models for semantic alignment and (2) the existence of an expressive meta-model and a manufacturing foundation ontology for providing a basis for ontology articulation.

The examples explored have shown the reconciliation of the GD&T semantics of hole features captured in disparate product representations. The same understanding can be

followed for the reconciliation of manufacturing process sequences, by using PSL structures [9]. Other mapping models can also be developed to target different reconciliation scenarios. For example, it is possible to formalise standards-based models, such as that of ISO Tolerance Band [9, 25]. Such a model can provide reusable semantic mapping concepts to facilitate the reconciliation of design and manufacturing features, by using the knowledge behind the dimensional tolerances carried by feature instances [9].

In the implemented approach, it is possible to employ the query tool in IODE to show the proof structure for verifying the occurrence of semantic mapping concepts. For example, it is possible to verify why the "classMappingRelation_018" (see Figure 10 (T')) holds between the classes "Drilled_Hole" and "Bored_Hole". The verification process uses the logic-based conditions that define semantic mapping concepts and is an essential facet of the rapid deductive reasoning mechanism used in the implementation platform. In fact, the successful retrieval of query results in the first place, already proves their verification at computational level.

In terms of performance during implementation, it was observed that the loading and saving of a mapping model containing around 100 semantic mapping concepts took about one minute to be performed using current available technologies. On the other hand, querying procedures for mapping discovery took less than or about 10 seconds to be processed. This clearly indicates that an attractive direction for ontology mapping/matching exists when weighted against other related methods which may take several minutes, hours or even days to complete a matching task [8].

From an industrial application viewpoint, the SMIF approach is aimed at complementing PLM in order to achieve an increased level of interoperability and integration amongst various stakeholders involved in collaborative product development. Figure 12 illustrates a possible configuration of the framework with respect to elements of wider design and manufacturing systems. Domain ontologies that derive from the Foundation Layer could be interfaced with CAE applications, for example, a CAD environment could be linked to a domain ontology that fully captures the semantics in solid modelling (I''). The KB associated with the domain ontology would be used as a repository for creating, storing, accessing and updating semantically-rich parts information.

In addition to this, the expressive semantics from PSL could be exploited for process planning activities (J''). Extensions to the framework aided through the set up of interfaces with PLM environments would potentially help support not only integration, but also desired level of semantic interoperability. The latter is an essential capability that is required for effectively sharing knowledge across multi-disciplinary teams in collaborative PLM (K''), and can be facilitated through the formalisation of relevant mapping models.



Figure 12: A possible configuration of SMIF within PLM

6 CONCLUSIONS

Ontology mapping/matching and verification techniques have so far not been sufficiently exploited in the field of product design engineering. This is partly due to the fact that ontology mapping/matching solutions tend to originate from the domain of computer science and, therefore, their benefits in product design engineering are acknowledged at a later stage. However, the logic-based approach investigated in this paper has documented an effective and accurate verification mechanism that has been purposely developed for the reconciliation of product and manufacturing representations.

The high degree of expressiveness of Common Logic-based structures has provided the possibility to establish resolvable and meaningful mappings across knowledge models. This helps support consistency during mapping and also avoids some of the drawbacks related to subjectivity in mapping transactions that involve extensive human intervention. The interpretation of the mapping correspondences, which is verified via tractable reasoning, is more rigorous when compared to other similar methods. For example, OWL-based reconciliation tends to be limited to mapping relationships such as "equivalentClass", "equivalentProperty" and "similarTo".

Finally, it is still required to conduct test cases based on comprehensive industrial scenarios. These scenarios would bring considerable value to the applicability of the proposed framework within industrial settings. Possible case studies, for example, from the aerospace and automotive industries would convey the breadth of product lifecycle concepts required for further testing and improving the framework components. Our current work on Interoperable Manufacturing Knowledge Systems (IMKS) intends to provide imminent extensions to this framework.

ACKNOWLEDGEMENTS

The work developed in this paper has been supported through a research studentship funded by the Wolfson School of Loughborough University. We also wish to thank the EPSRC, who are funding the majority of our work on "Interoperable Manufacturing Knowledge Systems" (IMKS) under project 253 of the Loughborough University Innovative Manufacturing and Construction Research Centre.

REFERENCES

[1] Subrahmanian, E., Rachuri, S., Fenves, S.J., Foufou, S. and Sriram, R.D., (2005), "Challenges in Supporting Product Design and Manufacturing in a Networked Economy: A PLM Perspective", *Proceedings of the International Conference on Product Lifecycle Management*, pp. 495-506.

[2] Kugathasan, P. and McMahon, C., (2001), "Multiple Viewpoint Models for Automotive Body-In-White Design", *International Journal of Production Research*, Vol. 39, No. 8, pp. 1698-1705.

[3] Gunendran, A.G. and Young, R.I.M., (2006), "An information and knowledge framework for multi-perspective design and manufacture", *International Journal of Computer Integrated Manufacturing*, Vol. 19, No. 4, pp. 326-338.

[4] Hameed, A., Preece, A. and Sleeman, D., (2004), "Ontology Reconciliation". In: Staab, S. and Studer, R., (Eds.), "Handbook on Ontologies, International Handbooks on Information Systems", pp. 231-250, Springer-Verlag, Berlin.

[5] Das, B., Cutting-Decelle, A.F., Young, R.I.M., Case, K., Rahimifard, S., Anumba, C.J. and Bouchlaghem, N., (2007), "Towards the Understanding of the Requirements of a Communication Language to Support Process Interoperation in Cross-Disciplinary Supply Chains", *International Journal of Computer Integrated Manufacturing*, Vol. 20, No. 4, pp. 396-410.

[6] Cutting-Decelle, A.F., Dubois, A.M. and Dubois, J.E., (2002), "An Information System for the Building Industries: A Communication Approach Based on Industrial Components", *Data Science Journal*, Vol. 1, No. 2, pp. 257-270.

[7] Pouchard, L., Ivezic, N. and Schlenoff, C., (2000), "Ontology Engineering for Distributed collaboration in Manufacturing", *Proceedings of the AlS2000 Artificial Intelligence and Simulation Conference*, Tuscon, Arizona, USA.

[8] Shvaiko, P. and Euzenat, J., (2008), "Ten Challenges for Ontology Matching", *Proceedings of the 7th International Conference on Ontologies, Databases and Applications of Semantics*, Monterrey, Mexico.

[9] Chungoora, N. and Young, R.I.M., (2010), "A Framework to Support Semantic Interoperability in Product Design and Manufacture", Ph.D. Thesis, Loughborough University, Loughborough , UK.

[10] Chungoora, N. and Young, R.I.M., (2010), "A Framework to Support Semantic Interoperability in Product Design and Manufacture", Accepted for publication in the 20th CIRP Conference, Nantes, France.

[11] Kalfoglou, Y. and Schorlemmer, M., (2003), "Ontology Mapping: The State of the Art", *Knowledge Engineering Review*, Vol. 18, No. 1, pp. 1-31.

[12] Noy, N.F. and Musen, M.A., (2003), "The PROMPT Suite: Interactive Tools for Ontology Merging and Mapping", *International Journal of Human-Computer Studies*, Vol. 59, pp. 983-1024.

[13] Euzenat, J. and Shvaiko, P., (2007), "Ontology Matching", Springer-Verlag. Berlin Heidelberg.

[14] Liping, Z., Guangyao, L., Yongquan, L. and Jing, S., (2007), "Design of Ontology Mapping Framework and Improvement of Similarity Computation", *Journal of Systems Engineering and Electronics*, Vol. 18, No. 3, pp. 641-645.

[15] Maedche, A., Motik, B., Silva, N. and Volz, R., (2002), "A MApping FRAmework for Distributed Ontologies", *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management, Ontologies and the Semantic Web*, LNCS, Vol. 2473, pp. 235-250.

[16] Stumme, G. and Maedche, A., (2001), "FCA-Merge: Bottom-Up Merging of Ontologies", *Proceedings of the 7th International Conference on Artificial Intelligence*, Seattle, WA, pp. 225-230.

[17] Highfleet Inc. (formerly Ontology Works Inc.). Accessible at: http://www.highfleet.com

[18] Delugach, H.S., (2005), "Common Logic in Support of Metadata and Ontologies", PowerPoint Presentation, *Open Forum 2005 on Metadata Registries*.

[19] Fenves, S.J., Foufou, S., Bock, C., Sriram, R.D. (2005), "CPM: a Core Product Model for Product Data", *Journal of Computing and Information Science in Engineering*, 5: 238-246.

[20] ISO 10303-224, (2006), STEP - Mechanical Product Definition for Process Planning Using Machining Features.

[21] ISO 18629, (2005), Industrial Automation Systems and Integration - Process Specification Language.

[22] Dani, S., Harding, J.A., Case, K., Young, R.I.M., Cochrane, S., Gao, J. and Baxter, D., (2006), "A Methodology for Best Practice Knowledge Management", *Proceedings of the IMechE, Part B: Journal of Engineering Manufacture*, Vol. 220, No. 10, pp. 1717-1728.

[23] Rabe, M. and Gocev, P., (2008), "Semantic Web Framework for Rule-Based Generation of Knowledge and Simulation of Manufacturing Systems", In: Mertins, K.,

Ruggaber, R., Popplewell, K. and Xu, X., (Eds.), "Enterprise Interoperability III: New Challenges and Industrial Approaches", pp. 397-409, Springer-Verlag London Ltd., UK.

[24] Lin, H.K., Harding, J.A. and Teoh, P.C., (2005), "An Inter-Enterprise Semantic Web System to Support Autonomy and Conflict Moderation", *Proceedings of the IMechE, Part B: Journal of Engineering Manufacture*, Vol. 219, No. 12, pp. 903-911.

[25] ISO 286-2,1988, Tables of Standard Tolerance Grades and Limit Deviations for Holes and Shafts.

[26] ISO/IEC 24707, (2007), Information Technology – Common Logic (CL): A Framework for a Family of Logic-Based Languages.