Developing Ontologies for Collaborative Engineering in Mechatronics

Violeta Damjanović, Wernher Behrendt, Manuela Plößnig, and Merlin Holzapfel

Salzburg Research, Jakob Haringer Strasse 5/II,
5020 Salzburg, Austria
{violeta.damjanovic, wernher.behrendt, manuela.ploessnig, merlin.holzapfel}@salzburgresearch.at
http://www.salzburgresearch.at

Abstract. Creating a coherent set of ontologies to support a collaborative design process amongst different firms which develop mechatronic products is a challenge due to the semantic heterogeneity of the underlying domain models and the amount of domain knowledge that needs to be covered. We tackle the problem of semantic heterogeneity by employing the DOLCE foundational ontology and by aligning our models to it. We approach the problem of scale, i.e. the amount of knowledge modeled by keeping the models at a descriptive level which is still granular enough to connect them with domain and task specific engineering tools. In order to manage the complexity of the modeling task we separate the models into the foundational layer, the mechatronic layer consisting of three domain ontologies, one process model and one cross-domain model, and the collaborative application layer. For the development process, we employ a methodology for dynamic ontology creation, which moves from taxonomical structures to formal models.

1 Introduction

The mechatronic engineering process covers an interdisciplinary combination of different domains comprising of mechanical engineering, electrical engineering, and software engineering. For each of these engineering domains there exist diverse knowledge models, mostly in the form of documents or glossaries, but hardly as comprehensive ontologies. Furthermore each domain covers a specific mechatronic field, so that the intersection of knowledge models between these different engineering domains remains relatively small.

The focus of the ImportNET[1] project lies in this intersection, specifically in the collaboration of the three mainstream mechatronic domains, i.e. mechanical, electrical, and software engineering. For this reason, the process of ontology modelling in ImportNET is considered from two perspectives:

Firstly - in the research perspective - we employ a methodology for dynamic creation of ontologies (i.e. moving from less formalised models to more rigorous models).

[1] The ImportNET project is co-funded by the European Commission within the Sixth Framework Programme under Contract 033610, in the area of ICT for Networked Businesses.

E. Franconi, M. Kifer, and W. May (Eds.): ESWC 2007, LNCS 4519, pp. 190–204, 2007.

Using the DynamOnt methodology [1] we model the reference ontologies for mechatronics on the basis of the DOLCE foundational ontology. This includes a generic mechatronic process model which will also be used to describe the usage scenarios for the modeling.

The second perspective concerns the actual use of ImportNET tools: there needs to be a methodology to modify the reference ontologies in order to adapt them to the requirements of concrete companies and their products. The reference ontologies must be tailored to the requirements of the actual, planned collaboration. This will be done by the Ontology Integration Tool (OIT) which allows to modify and to expand the reference ontologies.

The paper describes the early stages of work in a European research project and addresses collaborative design processes that are used for development of mechatronic products as follows: The introduction section gives a brief overview of the state of the art and existing research gaps in collaborative engineering, mechatronic engineering and mechatronic domain modeling. Section 2 introduces the ImportNET approach to mechatronic domain modeling. This section firstly discusses the ontology landscape in ImportNET, and then illustrates the alignment of the mechatronic ontology with the DOLCE foundational ontology. Furthermore the DynamOnt methodology, which is used for the evolutionary creation and development of the mechatronic ontology, is explained in more detail. Section 3 discusses the main objectives of ImportNET, the possible system architectures, as well as usage and validation scenarios. Preliminary conclusions are drawn in Section 4.

1.1 State of the Art and Research Gaps in Collaborative Engineering

In recent years, collaboration not only between engineers but also across organisational boundaries has become a key research issue for the development of flexible engineering processes. Collaborative engineering aims at providing the main concepts, solutions, as well as technologies for development of products by multiple engineering teams. We found the following main research challenges and gaps in the domain of collaborative engineering:

- technical aspects: Web-based electronic design environments; architectures and technologies for knowledge sharing; standards for exchange formats/protocols; security aspects;
- social aspects: handling multi-cultural issues in collaborative design; knowledge sharing; collaborative learning; collaborative engineering; distributed engineering work; social aspects of collaboration teams;
- organizational and economic aspects: benefits of using collaboration approaches; validation scenarios.

At the same time, there is a number of unsolved problems from the industrial perspective, including application integration e.g. how can Web Services contribute to closing this gap?; knowledge integration e.g. how can Semantic Web technologies contribute?; and process integration e.g. how can approaches like Enterprise Modeling answer to this challenge?

1.2 State of the Art and Research Challenges in Mechatronic Engineering

Mechatronic engineering is one of the most recent branches of engineering and it has increasing impact on many sectors of the economy and on society overall. The competitive use of mechatronic engineering will soon require more model-driven development using design repositories of mechatronic components. We have found two notable metamodels which address this issue: Thramboulidis describes a four-layer model of Integrated Mechatronics distinguishing mechanical, resource, application and mechatronic layers [2]. The model is the basis for ”Archimedes, a system platform that supports the engineering through a methodology, a framework and a set of tools to automate the development process of agile mechatronic manufacturing systems“ [2]. The problem of ontological modelling was addressed by Yoshioka [3] in a layered knowledge structure for the Knowledge Intensive Engineering Framework (KIEF). They also introduce the concept of plug-in models to specialise and refine the metamodel into concrete models. Their paper indicates that there is at least a proof-of-concept prototype in which some of the proposed concepts are validated. Unfortunately, the actual implementation is not in the public domain. Each of the two frameworks has a particular angle: Thramboulidis focuses on the mechatronic process whereas Yoshioka emphasises the modelling of mechatronic artefacts. Both models will have to be considered as frameworks for our collaboration-centered approach to mechatronics.

1.3 State of the Art and Research Challenges in Mechatronic Domain Modeling

Ontological engineering covers a whole range of topics such as the basic philo sophical and metaphysical issues as well as knowledge representation formalisms, me thodology for ontology development, business process modelling, commonsense knowledge, systematisation of domain knowledge, Internet information retrieval, standardisation, evaluation, and many more [4].

If we put ontological engineering in the context of other disciplines, then many similarities and analogies arise. They allow us to make connections between ontological engineering and the other disciplines, to bridge potential comprehension gaps, and to shed a different light on already known concepts and practices. For example, when applying the Unified Modeling Language (UML) to a mechatronic system it turns out that some additional concepts are needed to model the mechatronic system [5]. Such concepts can be added by introducing stereotypes, e.g. a special stereotype called the Function Block Adapter (FBA) is described in [5]. The FBA stereotype can be used to specify the mapping from UML signals to the function block signals.

2 ImportNET Approach to Mechatronic Domain Modeling

A review of the literature about mechatronics rapidly results in a number of definitions, each of which emphasises a slightly different aspect of the mechatronics concept, ranging from design to precision engineering and from sensors to actuators [6]. Most of the definitions do manage to agree that mechatronics is concerned with the integration of its core engineering themes to generate novel technological solutions in the form of products and systems whose functionality is integrated across those core technologies.

The design of an ontology for mechatronics can be approached using a variety of scientific methods, such as the following paradigms [7]:

- empirically-based research (cognitive models),
- axiom-based research (computational models); and
- conjecture-based research (computational models):
  - conjectures based on an analogy with cognitive processes; and
  - conjectures based on an analogy with computational processes.

Empirically-based research involves the development of experimental studies of designers that result in cognitive models of designing. Axiom-based research involves the identification of a set of axioms and their consequences to derive a logic-based computational model of designing. Conjecture-based research involves an analogy between a cognitive or computational process that leads to a computational model specific to designing.

The approach taken by ImportNET is to move from a cognitive model to a computational model, with the help of a foundational ontology which could be seen as a compromise between cognitive conjectures (the concepts of the ontology) and axiom-based computational models (the axiomatic framework defined by the DOLCE foundational ontology).

2.1 Ontology Landscape in ImportNET

Ontologies provide the vocabulary for referring to the terms in a subject area, as well as the logical statements that describe what the terms mean, how they are related to each other, as well as the rules for combining terms and relations to define extensions to the vocabulary.

Figure 1 provides a landscape of reference ontologies employed in ImportNet. The DOLCE ontology represents the foundational layer which gives us a useful structure for building novel knowledge based architectures. Aligned to DOLCE, we place the domain ontologies for mechanical, electronics and software engineering. The new cross-domain engineering ontology is built as a result of the integration of these contributing ontologies, while the mechatronic engineering lifecycle ontology has to be linked ultimately, to distributed service execution and orchestration processes.

The ImportNET ontologies are created in support of a collaborative engineering process for developing mechatronic products. The process for development of the mechatronic products requires some ontology integration and configuration based on the overall set of ontologies. The ontology landscape and its configuration via the Ontology Integration Tool (OIT) is shown in Figure 1. The resulting Collaboration Ontology is a meaningful subset of concepts from the ontology landscape. Any collaboration between organisations developing a specific product will be based on such a specialised collaboration ontology. The design and detailed functionality of OIT are outside the scope of this paper.

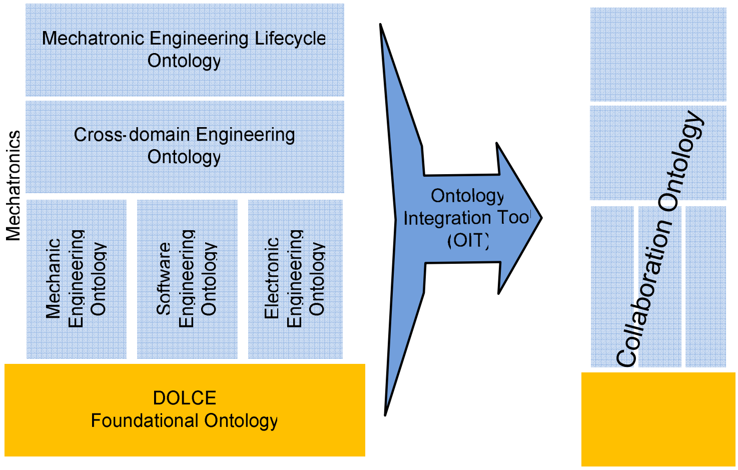

Fig. 1. Ontology landscape and configuration of a collaboration ontology

The design of the cross-domain engineering ontology is considered to be an essential theme for mechatronics since it attempts to bring together concepts and ideas in relation to a product or system [6]. Furthermore, the design of a flexible mechatronic engineering lifecycle ontology to support the collaborative development of mechatronic products amongst various communities of practice and virtual organizations is the main challenge in ImportNET.

A partial taxonomy of mechatronic ontologies is represented in Tables 1-5.

Table 1. A partial taxonomy of the mechanical engineering ontology

Criteria	Explanation
Spatial	Spatial description of mechanical components
Composing	Aggregation / assembly
Properties	Physical properties, e.g. liquid
Process	- domain specific workflow - mechanical behaviour of components, e.g. rotation or movement along a trajectory
Role	Roles of agents in the domain of mechanical engineering, e.g. material stress tester
Methods	Methods of mechanical engineering

Table 2. A partial taxonomy of the electronic engineering ontology

Criteria	Explanation
Spatial	Spatial description of electronic components
Composing	Aggregation / assembly
Properties	Physical properties
Process	- Domain specific workflow - Electro magnetic behaviour
Role	Roles of agents in the domain of electronic engineering
Methods	Methods of electronic engineering

Table 3. A partial taxonomy of the software engineering ontology

Criteria	Explanation
Functions	Architecture of the runtime environment, hardware drivers
Composing	Aggregation / assembly
Properties	Description of design, documentation, code, APIs…
Process	- Software life cycle - Behaviour of software components
Role	Roles of agents in the domain of software engineering
Methods	Methods of software engineering

Table 4. A partial taxonomy of the mechatronic engineering lifecycle (process) ontology

Criteria	Explanation
Composing	Sub-processes at different levels of granularity, requiring input/output parameters to be modeled at corresponding levels of detail
Properties	Characterising different instantiations of a process model (e.g. waterfall, V-model, etc), order of sub-processes, duration, pre- and postconditions
Role	Roles of agents in particular those engaged in coordinating and resolving conflicts between the engineering domains
Methods	E.g. conflict resolution between roles

Table 5. A partial taxonomy of the Cross-domain Ontology

Criteria	Explanation
Spatial	Runtime environment, hardware drivers
Composing	Aggregation / assembly
Process	Electro magnetic behaviour, software/ hardware execution…
Role	Union of roles defined in the other domains
Methods	E.g. conflict resolution between roles

Current work is addressing the relationship between the initial taxonomies and the frameworks proposed by Thramboulidis [2] and Yoshioka [3]. One of the main issues in combining the knowledge of these other models with ImportNET is that once we have made a commitment to a foundational ontology we need to also align external models to that foundation. For example, each mereological element of an external ontology needs to be mapped into the corresponding primitives of the foundational ontology. Whether or not there is a specific ontological bias in any of the external models can only be determined once we have access to the full models.

2.2 Ontology Alignment to the DOLCE Foundational Ontology

The Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) was originally developed in the EU WonderWeb project [8] and has been extended in a number of other projects since then. The design philosophy of DOLCE is modularity in order for ontology projects to be able to pick and choose thus making only as much ”ontological commitment“ as needed. The typical process of developing an ontology is then to either ”align“ existing knowledge models to the DOLCE model or to develop the ontology from scratch, by using the conceptual primitives defined by DOLCE.

Despite the ambition to capture some ”common sense“ DOLCE constitutes a strictly formal approach to ontology modeling, which is a necessary condition if we want computational services or agents to make autonomous use of the ImportNET knowledge models while remaining ”accountable“ for their activities. Such semantic accountability is an important requirement for future work spaces where some of the decision making in cross-organisational processes will be delegated to machines and where there will be a need at least for boundary conditions to be defined explicitly in order to safeguard against unwanted behaviour of partly autonomous systems. Furthermore, the axiomatisation is a prerequisite for any logic based inferencing done by such machines.

The fundamental difference between current ”semantic“ terminologies as used in annotations and ”proper“ semantic models as envisaged for the ImportNET Semantic Application Server (SAS) is that the latter will have to implement a partly autonomous inference module in order to manage the cross-organisational work processes, which will be context-sensitive to the mechatronic design artefacts which will be exchanged between the engineers (i.e. the users of the system).

Since ImportNET will focus on cross-organisational processes it will be necessary to add the capability for modeling tasks to the basic model. The process of aligning the set of mechatronic ontologies to DOLCE is shown in Figure 2. In the recent EU METOKIS project, DOLCE was extended by an ontology called ”Descriptions and Situations“ (D&S), which includes a representation language for tasks or processes [9]. D&S shows its practical value when applied to ontology design patterns for (re)structuring application ontologies that require contextualization [10]. Figure 3 represents the process of aligning the mechatronic ontologies with respect to the basic categories of DOLCE, as well as using of Semantic Web Services approach to support the collaborative design process.

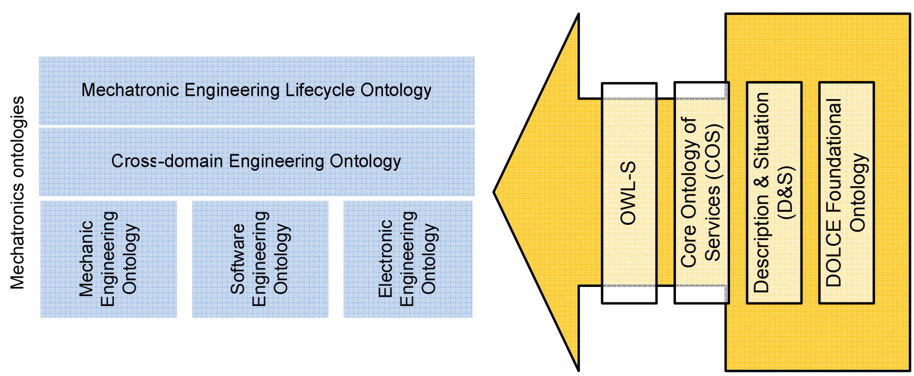

Fig. 2. The DOLCE foundational ontology is extended by the D&S module. Instead of directly aligning OWL-S to D&S, a Core Ontology of Services (COS) is developed and OWL-S is aligned to the COS ontology [10]. COS tries to fill the epistemological gap between the foundational ontology and OWL-S, and also it can be reused to align other Web Service Description Languages as well. Furthermore, COS ontology is used to align the set of mechatronic reference ontologies.

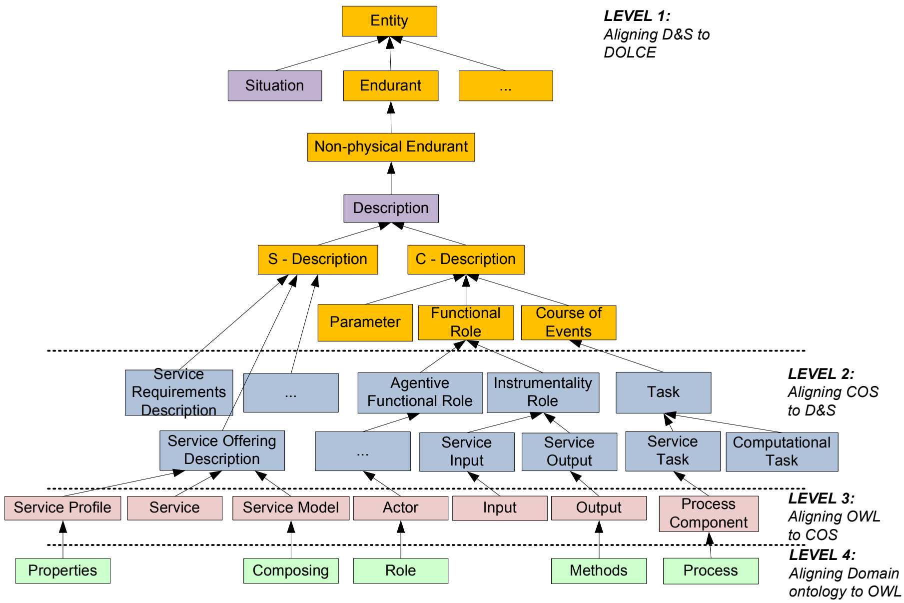

Fig. 3. Indirectly aligning mechatronic ontologies to the DOLCE foundational ontology

To summarise, the extended DOLCE foundational ontology for which a full implementation in OWL-DL exists has been chosen as the working hypothesis from which the modeling of the ImportNET ontologies start. Achieving such a combined representation in the area of mechatronics would be a significant result because to our knowledge, no other foundational model has a comparable degree of coherence and formalisation.

2.3 Methodology for the Development of the Mechatronic Ontology

Ontology development methodologies are intended to help with the complex process of ontology building and managing. They help knowledge engineering projects to successfully reach the main goals in time, especially when it comes to knowledge sharing in dynamic environments due to frequent changes of user needs.

There are two general ontology engineering approaches, *centralized* and *decentralized* methodologies. On-To-Knowledge (OTK) [11] and METHONTOLOGY [11], [12] are mostly centralized, while DILIGENT [13] and the recently proposed DynamOnt [1] methodology can be seen as decentralized and distributed approaches for ontology engineering where a community of ontology users and developers converges towards a shared view.

For the development of the contributing ImportNET ontologies, we use the DynamOnt methodology. The DynamOnt methodology enables the dynamic creation of ontologies based on communication and experience exchange amongst different communities of practice - in our specific case those communities which are concerned with the development of mechatronic products.

The DynamOnt process model integrates elements of known knowledge and ontology-engineering methods in order to produce an overall methodology for engineering of knowledge-based systems. In detail the DynamOnt model comprises the following phases [1]:

- **Identify the problem** – domain experts (users) could describe the situation when the problem occurs or they have ideas to solve the problem;
- **Structure the problem** – a broader discussion with domain experts (users) and the description of user scenarios would help to structure the problem in order to get a broader view of the topic;
- **Identify concrete purpose and scenarios** – the focus is a mutual understanding of the project goals. A guideline based on a three dimensional matrix is proposed to classify ontologies along the properties scope (stability of knowledge models and interoperability on semantic level), expressiveness (complexity and costs), and acceptance (market success and collaboration);
- **Identify main concepts of domain/subject matter** – based on user scenarios, existing documents and knowledge models a list of domain concepts, roles and tasks will be created;
- **Create non-formal models** – the already defined concepts, roles and tasks will be interrelated through attributes and relations. This will be supported by guided questions;
- **Create formal models (knowledge design)** – the classification according to the expressiveness dimension of the three dimensional matrix helps to decide which parts of the ontologies has to be formalised to a certain degree. Based on the non-formal model and maybe other available models, a conceptual (formal) model will be defined and the output will be machine readable (e.g. OWL, RDFS, XML);
- **Create acceptance (community design)** - the acceptance within the main user communities (e.g. developers, the domain experts, external user communities of the system) is an important factor for the success of the model and the system. The acceptance can be raised by trainings (e.g. workshops) and by adapting existing business processes according to inputs of the resulting formal model;
- **Create system (software design)** – based on software engineering methods and techniques the software will be specified and designed;
- **Implement Target System** - the scope of this phase is to provide a fully developed knowledge-driven application.

In the formalisation steps, DynamOnt uses the following ontological design patterns (based on DOLCE) to guide domain experts in creating conceptualisations of their domain knowledge [1]:

- the Participation pattern;
- the Description-Situation pattern;
- the Role-Task pattern;
- the Design-Artefact pattern;
- the Agent-Activities pattern;
- the Information-Object pattern.

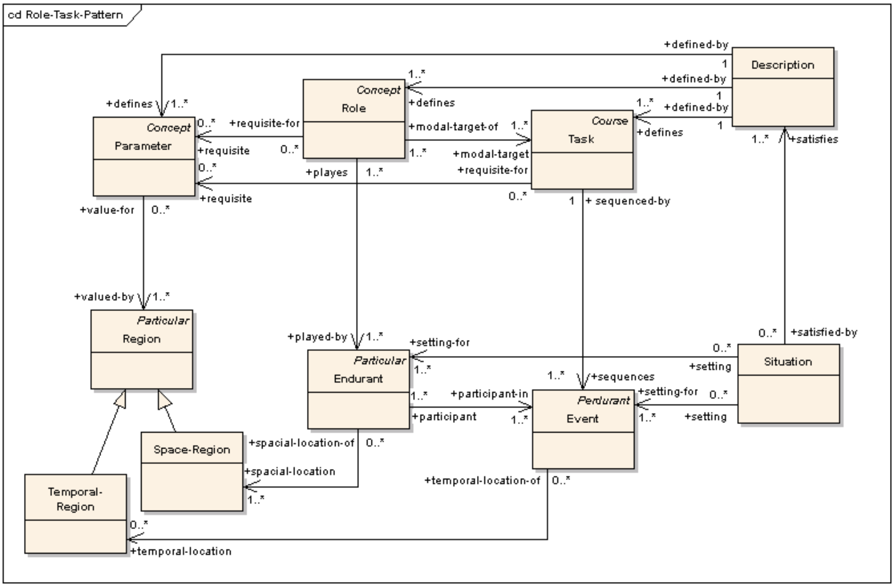

Fig. 4. Role-Task ontological design pattern defined in DynamOnt using DOLCE concepts

Each of these patterns acts as a modelling template to describe how agents in various roles, participate in situations and use information objects for communication. The use of these patterns is similar to the design patterns in object oriented programming and it should lead to a more homogeneous way of modelling intelligent agents, roles and activities in any environment. Figure 4 shows the Role-Task pattern, which is defined in DynamOnt methodology by using the DOLCE concepts.

One of the early lessons of our ontology work is that a common set of knowledge engineering methods would be desirable. Methontology appears to be the most straightforward approach to semiformal modelling, but lacks the foundational rigour of DOLCE which is better supported by the ontological patterns of DynamOnt. Methontology on the other hand, offers good ways to express axioms and rules which are absolutely necessary constructs for designing real-world semantic applications.

3 Objectives, ImportNET System Architecture, Usage and Validation Scenario

The ImportNET project is addressing on the one hand, the issue of creating a support environment for virtual enterprises in cross-domain engineering and on the other hand the problem of cultural differences and misunderstandings which may lead to communication failures between engineers who try to collaborate with each other.

3.1 Objectives and Initial Findings

The technical approach is to first create the collaboration environment by integrating the knowledge models of the three engineering domains and by creating a layer of supporting middleware to integrate existing engineering tools (CAE, CAD/CAM, CASE). In parallel, a knowledge base of intercultural communication problems is being developed and the communication flow between engineers is analyzed, along the mechatronic product life cycle. The communication will be modeled explicitly, in the collaboration ontology which specialises the domain ontologies for a specific collaboration between some firms developing some defined product. The intercultural knowledge base will be indexed in such a way as to enable the triggering of ”warnings“ when there is a likelihood of a misunderstanding occurring in a communication act along the lifecycle.

For the integration of the engineering tools into a collaborative lifecycle support environment it will be necessary to create ”wrappers“ which translate the proprietary or otherwise incompatible data formats into semantically comparable intermediate representations. To automate some of this translation process an Intelligent Adapter Generation Tool (IAGT) is envisaged. This tool will use compiler-compiler techniques to specify the semantic relationships between a proprietary model and the intermediate representation and to create from this specification, two-way translators which can be integrated into the collaboration environment.

The integration of the three domains has already been described: we use DOLCE as a foundational ontology and specialize the D&S module to the needs of modeling processes in cross-domain engineering. In order to make it easier for organisations as well as for technology integrators, to specify a workflow for a new collaboration, we make use of the OIT. This tool will offer semantic templates (“ontological patterns“) to the integrator, which can be specialised for the needs of a new collaborative engineering project.

3.2 ImportNET System Architecture and Issues Around Semantic Modelling

The system comprises of a knowledge based back-end called SAS (Semantic Application Server), and a client front-end application called MDET (Multi Domain Engineering Tool). The MDET offers different engineers their preferred view of the overall system and it mediates potential misunderstandings by being aware of the communication acts between the participants of the collaboration. One of the roles of MDET will be to mediate between mechanical engineering views (which are typically 3D) and electronic views (normally 2D). The challenge lies in merging the internal representations of external engineering design tools (eCAD and mCAD) into a common one with uniform semantics. This resolution will be done in the SAS with the help of the tool adapters (i.e. semantic wrappers) constructed with the help of the IAGT. The SAS plays the role of a semantics-based middleware which connects the external tools to the ImportNET communication and collaboration processes. We have identified three issues that such a system needs to address: a) the role of inference support; b) the need for semantic web services; c) the degree of interoperation between current engineering tools.

Role of inference support: current CAD tools are based on object-oriented, often proprietary database back-ends. Similarly, even most of the open research systems in the field of engineering are based on object-oriented data models. Any semantic interoperation approach is faced with the dilemma that one has to either replicate the data in a Semantic-Web enabled knowledge base in order to use inference engines or, to reimplement some inferencing capability on top of the existing OO datastore. This is a general problem facing Semantic Web applications when they need to interoperate with software in the commercial domain.

The need for semantic web services in ImportNET: the implementation architecture of ImportNET could be envisioned as an open, yet collaborative lifecycle support environment in which different Semantic Web Services can find each other automatically. This kind of ImportNET system architecture could be based on the Web Service Execution Environment (WSMX) core architecture, which enables discovery, selection, mediation, invocation and interoperation of Semantic Web Services [14]. However, it is not yet clear whether this kind of spontaneous semantic service integration is really needed for ImportNET, because the philosophy behind the system is a planned collaboration between known organisations and systems.

Degree of interoperation between current engineering tools: we see a major hurdle for the envisaged system still, in the complex yet proprietary solutions that are currently prevalent in engineering domains. This necessitates firstly, the approach of building an external semantic application server with its associated problem of inference engines versus object-model. Secondly, it also bears the danger of "research at a dead end" because we cannot research semantically interoperable models when the actual target application software is *designed to hinder or defeat*, interoperation, for reasons of market protection. One such example is that object structures are based on OIDs which are generated afresh each time a design is loaded and there are only limited ways of exchanging typed structural (schematic) information between different tools. This leads to a need for effectively reverse-engineering some of the functionalities of the target tools which is neither a worthwhile research question nor strictly legal in some cases. There is, however, an interesting side effect to this issue: semantic modelling points directly at methods by which commercial players are trying to protect their intellectual property and market share. The legal system may one day employ semantic modelling to determine what kinds of protection are fair and which methods of protection are detrimental to a competitive market.

3.3 Usage and Validation Scenario for Collaborative Mechatronic Design

As described above mechatronic engineering deals with collaboration across different domains. Each of these engineering domains is well supported by a range of engineering tools which cover at most the domain itself, but typically focus on a specific aspect e.g. the design of physical artefacts or the specification of automated tests for an electronic device. The focus of ImportNET and consequently of the case studies in ImportNET is on the design phases of the mechatronic lifecycle and the cross domain cooperations. The mechatronic life cycle coordinates the different tasks of the mechatronic engineering domains and the engineering tasks of one domain often influence the engineering tasks of another domain. As a result, the precise hand-over points of these tasks are sometimes not clear and coordination conflicts may occur. Against that background several aspects have to be studied and validated through use cases:

– During design many documents (e.g. output of CAD tools) need to be exchanged between the mechanic and the electronic engineering domains. Most of the documents are in a proprietary format and are therefore not easily imported by other

tools. Based on known exchange formats such as DXF[2] and STEP[3] ImportNET analyses where data can be automatically exchanged during cross-domain collaboration.
- Designing a mechatronic product involves engineering experts from different domains and conflicts can occur for several reasons. Often conflicts have simply a factual basis where e.g. a mechanic and an electronic engineer have to clarify technical issues. The mechatronic design process comprises in these cases the co-ordination of cross-domain issues with respect to spatial, temporal or causal relationships. The coordination between mechanical and electronic engineering can be very intricate because of interactions in space and in behaviour (e.g. thermal or electromagnetic dependencies).
- Companies are often from different countries and conflicts can also be caused by different cultural backgrounds (e.g. different time conceptualisations or communications habits). This may lead to misunderstandings when messages or behaviours are being interpreted in different ways.

ImportNET is developing two use cases where engineering experts (mechanic, electronic, software, testing) from different companies and different countries are involved. The basis for the description of the use cases is a general mechatronic life-cycle model which will be tailored firstly to the needs of the participating companies and secondly to requirements of the target mechatronic product which will be designed between these companies.

4 Conclusions

Creating a cross-domain engineering environment requires - irrespective of whether one uses a Semantic Web based approach or not - some understanding of the underlying domains and also an understanding of the maturity of the field. In the case of mechatronics, we found a mixed situation: each of the domains has relatively mature software tools for the design of new artefacts and the domain of manufacturing overall, has relatively mature standards such as STEP for the description of products. What is clearly missing is the integration of the design tools along the product life cycle and in the case of cross-domain engineering, the ability to transform the representations of one design tool into semantically equivalent representations for the perspective of a corresponding tool in the other engineering domain. Initial interviews with senior engineers revealed that up to a third of the development cost originates in the area of testing and that there is large scope for improvement in this phase of the product life cycle.

A first analysis of candidate ontologies revealed a good number of conceptual models not only at varying levels of generalisation but also with varying angles on the purpose of the system and hence, the choice of concepts.

There are at least three challenges in the project: defining a coherent set of partial ontologies, integrating a knowledge base of intercultural communication conflicts into the workflow model and integrating a Semantic Web Service architecture with the

[2] Drawing Exchange Format.
[3] STandard for the Exchange of Product model data.

process model of the mechatronic domain to ensure interoperation during the mechatronic design phase. In this paper, we have described our approach and methodological choices with respect to the development of the ontologies and we have outlined the implementation architecture for the case of Semantic Web Services. We have not addressed the integration of the intercultural issues yet. Another issue which is still to be addressed concerns the suitability of DOLCE as a foundational ontology for domains such as mechanical engineering and electronics. The current view is that DOLCE is a good choice as long as we do not need to engage in ”deep modelling“ i.e. the modeling of causes and effects or of constraints in physical systems. However, this begs the question whether current Semantic Web modelling is capable of integrating well, with any models that offer ”analogue“, i.e. numerical or function-based simulation of system behaviour.

Acknowledgments. This work has been supported by the ImportNET project (IST-2006-033610). We would like to thank the referees for their helpful comments. We also acknowledge fruitful discussions with our colleagues Miklós Szőts, Alexander Mahl, Oliver Hornberg, Ron Jamieson and Rupert Westenthaler.

References

1. DynamOnt Deliverable 201: First specification of Methodology and Workbench for Dynamic Ontology Creation (2006)
2. Thramboulidis, K.: Model Integrated Mechatronics – Towards a new Paradigm in the Development of Manufacturing Systems. IEEE Transactions on Industrial Informatics, Vol. 1, No. 1 (2005) 54-61
3. Yoshioka M.: Physical Concept Ontology for the Knowledge Intensive Engineering Framework. In Advanced Engineering Informatics, Vol. 18, No. 2 (2004) 95-113
4. Devedzić, V.: Understading Ontological Engineering. Communications of the ACM, Vol. 45, No. 4 (2002) 136–144
5. Heverhagen, T., Tracht, R.: Using Stereotypes of the Unified Modeling Language in Mechatronic Systems. In Proceedings of the First International Conference on Information Technology in Mechatronics, ITM'01, Istanbul, Turkey (2001) 333–338
6. Bradley, D.: What is Mechatronic and Why Teach It? International Journal of Engineering Education (2004) Online: http://findarticles.com/p/articles/mi_qa3792/is_200410/ai_n10298146/g_1 (Last access: 2007-03-13)
7. Gero, J.S., Maher, M.L.: A Framework for Research in Design Computing. In Martens, B., Linzer, H., and Voigt, A. (eds), ECAADE'97, Osterreichischer Kunst und Kulturverlag, Vienna (1997) Online: http://people.arch.usyd.edu.au/~john/ publications/1997/ecaade/ index.html (Last Access: 2007-03-13)
8. Masolo, C., Borgo, S., Gangemi, A., Guarino, N.: Ontology Library. WonderWeb Deliverable D18 (2004) Online: http://wonderweb.semanticweb.org/deliverables/documents/D18.pdf (Last access: 2006-10-23)
9. Gangemi, A., Borgo, S., Catenacci, C., Lehmann, J.: Task Taxonomies for Knowledge Content, METOKIS Deliverable D07 (2004)
10. Mika, P., Oberle, D., Gangemi, A., Sabou, M.: Foundations for Service Ontologies: Aligning OWL-S to DOLCE, In Proceedings of the 13th International Conference on World Wide Web 2004 (2004) 563-572

11. Nagypál, G.: Methodology for Building SWS Ontologies in DIP. DIP Deliverable D3.11 (2005)
12. Fernández M., Gómez-Pérez, A., Juristo, N.: METHONTOLOGY: From Ontological Art towards Ontological Engineering. In Proceedings of the AAAI Spring Symp. Series, AAAI Press, Menlo Park, California (1997) 33-40
13. Sofia Pinto, H., Staab, S., Tempich, C.: DILIGENT: Towards a Fine-Grained Methodology for Distributed, Loosely-controlled and Evolving Engineering of Ontologies. In Proceedings of the ECAI 2004 (2004) 393-397
14. Zaremba, M., Moran, M., Haselwanter, T., Lee, H-K.: WSMX Architecture, WSMX Deliverable D13.4 (2005)