Proceedings of the ASME 2008 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2008
August 3-6, 2008, Brooklyn, New York, USA

DETC2008-49730

SEMANTIC ANNOTATION OF DIGITAL ENGINEERING RESOURCES FOR MULTIDISCIPLINARY DESIGN COLLABORATION

QZ Yang*
Singapore Institute of Manufacturing Technology
Singapore

Y Zhang
Stansfield College
Singapore

CY Miao
Nanyang Technological University
Singapore

ZQ Shen
Nanyang Technological University
Singapore

ABSTRACT

This paper introduces an ontology-based approach to annotating semantics of digital engineering resources. The aim is to enhance design knowledge sharing through semantic annotation to support streamlined collaboration in multidisciplinary consumer product development. Two issues are focused: how to specify the meaning of annotations with design ontology to ensure sharability of the annotation content; and how to represent annotations in neutral encoding formats to seek mutual understanding of the annotated semantics across multidisciplinary design teams and systems. Two use scenarios of semantic annotations in multidisciplinary design of consumer products are illustrated in the paper.

KEYWORDS

Semantic annotation, multidisciplinary product design, digital engineering resources, design collaboration, ontology

1. INTRODUCTION

Semantic annotation to design rationale and knowledge [1, 2] presents a big challenge, especially in multidisciplinary design, where the meaning of heterogeneous engineering resources must be shared across disciplines to enable seamless design collaboration. Take the consumer product design as an example. Its design collaboration typically involves mechanical, electrical, optical and software teams, design supporting disciplines, and outsourced design service suppliers. Each collaboration participant works on its particular engineering tasks for the common product goals. Discipline-specific design models, tools and processes are used to generate, utilize and manage design information in diversity of engineering resources. These resources may include CAD drawings, product data models, technical documents, engineering databases and software executables. However, most of the current computer support systems used in the consumer product design are mono-disciplinary and lack of the ability to understand, interpret and utilize semantics of the heterogeneous information from other systems. As a result, much of the rationale and implication of the domain-specific designs is stripped away in the design communication process from one discipline to another. This situation calls for the development of new methodologies and tools to enable design semantics sharing across disciplines in collaborative consumer product design.

This paper proposes an ontology-based approach to annotating semantics of digital engineering resources. The aim is to enhance design knowledge sharing through semantic annotations to support streamlined collaboration in multidisciplinary consumer product development. With this approach, annotation expressions are specified to explicitly capture the design characteristics implicated in disciplinary engineering resources. The concepts and relationships used in the annotation expressions are formalized with design ontologies. By populating the semantic annotation expressions with the design information and knowledge explicated and extracted from the multidisciplinary engineering resources or

* Author of correspondence, Phone: 65-67938348, Fax: 65-67916377, Email: qyang@simtech.a-star.edu.sg

from user-supplied semantic information sources, the design annotations are generated. As the semantics of the design annotations have been specified by formal concept classes and logic axioms in design ontologies, these design annotations can then support sharing of the meanings behind the resource documents, drawings or data models. To illustrate the feasibility of the proposed approach, two use scenarios are discussed for meaningful communication of design intent across disciplines; and for multi-CAD semantics sharing in collaborative consumer product development, respectively.

2. RELATED WORK

2.1 Engineering design annotation

Engineering design annotation has been used in communicating technical illustrations of designer's intention for centuries. Specifically, the textual annotations [19, 20] have been widely accepted as a means to capture, add and comment the information contents of engineering resources. However, creating textual annotations from engineering resources has been tedious and difficult to maintain with design changes. To overcome this, digital annotations have emerged. Digital annotations associate the annotation contents with the annotated design resources through links in a digital environment, so that the information contents of resources can be communicated more timely, clearly and efficiently. Most of the leading commercial CAD systems, as well as some research prototypes [3~6] have been developed with digital CAD model annotation capabilities. These systems enable designers to capture, retrieve and utilize the shared design information through CAD annotations. Despite the digital annotation technology has brought the added benefits to designers, it does not provide sufficient solutions to *semantics* sharing of multidisciplinary engineering resources.

Semantic annotation provides a new way, beyond textual annotation and digital link annotation, to produce and utilize annotations for sharing design intent in digital environments. It uses engineering ontologies to describe the semantics of annotations. As ontologies represent agreed upon views in design, they help eliminate ambiguity in interpretation of annotations. Annotation ontologies are usually modeled in formal and structural languages, such as RDF (Resource Description Framework) [7] and OWL (Web Ontology Language) [8]. The latter is used in the present study to describe the semantics of the annotated engineering resources.

2.2 Semantic annotation applications

Semantic annotation is most commonly used in annotating Web resources for search and discovery of Web contents. The basic idea is to annotate Web resources with semantic metadata provided by formal ontological models. The general annotation steps include: Web entity identification, entity disambiguation and annotation [9]. Through these annotation steps, the identified entities in Web resources are directly matched with the concepts in the ontological models to generate annotations. There exist numbers of semantic annotation frameworks and systems for creating Web resource annotations. Uren et al. [10] analyzed 27 existing Web annotation systems in a recent report. More semantic annotation environments, services, platforms and techniques were summarized in Reeve's study [11].

The semantic annotation technology has also been applied to the product design and manufacturing domains. Several EU projects, such as SEVENPRO [14], ATHENA [15] and INTEROP-NoE [16] have developed semantic annotation methodologies and systems to improve engineering knowledge sharing in networked enterprises. In another study, Kitamura et al. [12] proposed a functional ontology-based metadata schema for functional annotation of engineering designs in the Semantic Web. Their functional annotations can show designer's intentions behind the technical documents, thus facilitate efficient retrieval of the documents. Lin et al. [13] developed a semantic annotation framework for annotating process models. Four perspectives are studied in this framework: profile annotation, meta-model annotation, process model annotation and goal annotation in order to achieve semantic interoperability in enterprise process management.

While each of the methods above is suitable for a specific type of semantic annotation applications, such as for enterprise model interoperation or for acquisition, formalization and usage of product knowledge [14], some annotation principles presented can be applied to the multidisciplinary design annotation applications in consumer product development. Next section proposes such a method.

3. METHODOLOGY

The proposed semantic annotation approach consists of four steps:

- modeling design annotation expressions to define what to annotate using agreed common vocabulary;
- formalizing annotation semantics in these expressions with OWL ontologies;
- creating semantic annotations by instantiating the annotation expressions with the semantic information explicated from the existing resources, and enriched from external systems or from design teams' expertise (details in Section 3.3); and
- encoding annotations in OWL/RDF to enhance machine-interpretability and sharability of semantic annotations.

The following sections elaborate these four steps.

3.1 Modeling design annotation expressions

A design annotation expression is a representation of the annotation requirements from a particular design application, such as a CAD application or a sampling test application. It explicitly specifies the annotation content and structure for a selected design perspective to fulfill certain semantic annotation needs from that application. The design annotation expressions are modeled with annotation attributes and contextual constraints: 1) Attributes establish the meaning and information content of the annotation requirements. An attribute has an identifier and a type indicator. The identifier is taken from a commonly-agreed vocabulary and described with a concept class in design ontologies (in Section 3.2). Hence the identifier can specify the intended meaning of the attribute through semantic reference to the vocabulary and ontology. The type indicator indicates the stipulated data type of the attribute value. 2) Contextual Constraints contain a set of methods and relationships to specify the validity of the attribute values, scopes and types that an application system (e.g. a CAD or a product qualification test system) can support.

The annotation attributes and contextual constraints are organized in a unified structure using the following grammar:

<Attribute.Identifier(Attribute.TypeIndicator[:Contextual Constraint]), + > (1)

Using this structure as a template, design annotation expressions are defined. For example, the following annotation expression is instantiated from Expression (1) with five attributes and two contextual constraints. It is used to specify the identified design annotation needs for "inter-part relationship metadata" and "online product catalog" in one of our case studies.

<BehaviorCategory(String), BehaviorName(String: Unique), Description(String), Value(String), ExeType(Enum: dvb, java, cc, html)> (2)

As multi-domains, multi-views, and multi-resources are involved in creation and consumption of design annotations, semantic issues are becoming critical. To make the annotation contents understandable and reusable among collaborating agents (humans and systems), the start point is that any attribute identifiers in an annotation expression, such as *BehaviorCategory* and *ExeType* in Expression (2), must be taken from controlled vocabularies. Moreover, the terminologies/concepts in the vocabularies must be described in a formal, explicit and semantically-sound way.

3.2 Formalization of annotation semantics with OWL ontologies

A commonly-agreed vocabulary of domain concepts is a necessity for naming the attribute identifiers of annotation expressions. The aim is to remove ambiguity and misunderstanding to the shared use of the concepts in design annotations. This is done by the use of controlled vocabulary terminologies with their meanings and relationships being explicitly defined and agreed by all involved collaborating participants. However, how to make them to agree with the common vocabulary is not a topic of this paper.

In our approach, the controlled terminologies/concepts are collected through the annotation needs identification process. The process specifies what sharable information is required by a design application but missing from a given resource. It therefore needs to be added as a design annotation. Through this process, an initial set of annotation needs has been identified from the involved engineering applications, i.e., mechanical CAD, electrical and electronic CAD, optical CAD, qualification test, and collaborative design process management. The identified annotation needs are represented by additional metadata for semantic properties, CAD manipulations, reference links to external data sources, inter-part relationships and constraints, etc. The domain concepts used in the metadata representation are classified into two categories: property annotation concepts and behavioral annotation concepts. Property annotation concepts are used to enrich and explicate the *data semantics* of design resources through formal definitions of data meaning of the resources. Behavioral annotation concepts, on the other hand, are specified to enrich and explicate the *functional and executional semantics* related to design resources. This is done by formal definitions of functionality elements (e.g. capabilities of a CAD manipulation, requirements of a design constraint, etc). Fig. 1 depicts a partial classification hierarchy of the behavioral annotation concepts with a focus on the *Inter-Part Relationship* concepts needed in multidisciplinary consumer product design.

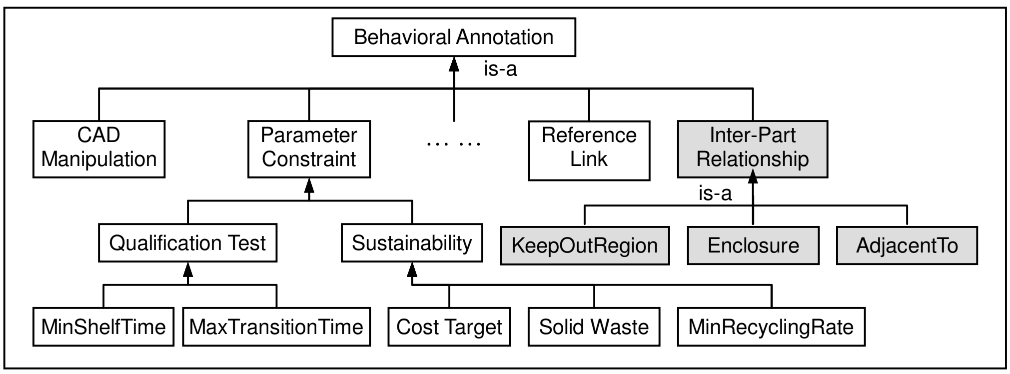

Fig. 1 A partial classification hierarchy for behavioral annotation concepts

The identified domain concepts are organized in a common vocabulary library. Meanings of the concepts in the library are specified explicitly. Basic inter-concept relationships include: (1) "*is-a*" between a sub-concept and its super-concept; and (2) "*part-of*" between a concept and its attributes. The vocabulary library stores and manages these domain concepts together with their intended meanings and inter-relationships.

To enhance the semantic description capability of the annotation vocabulary, OWL [8] ontologies have been developed for semantic annotation in multidisciplinary design. These design ontologies are intended to provide conceptual knowledge models for describing, understanding and sharing the semantics of design annotations across multiple CAD systems, product qualification test applications, and design process management systems in the current research. There exist different semantics sharing mechanisms, such as semantic transformation and semantic annotation. Here our semantics sharing mechanism used is by semantic annotation. As such, the mapping and inference between different annotation ontologies are not discussed in this paper. The following elaborates an example on how domain concepts in the annotation vocabulary are specified by OWL classes.

The sub-concept *AdjacentTo* in Fig. 1 is described by OWL ontological definitions as shown in Table 1. It indicates that the concept inherits four attributes from its super-concept *InterPartRelationship* in Fig. 1. It also specifies two additional attributes (i.e. *hasMcadPart* and *hasEcadPart*). By using OWL property constructs of *owl:ObjectProperty* and *owl:DatatypeProperty* [21], all the six attributes of the *AdjacentTo* concept class can be fully specified.

Table 1 OWL properties of *AdjacentTo* concept

Property		Domain	Range
At super-concept level for *InterPart Relationship*	owl:DatatypeProperty hasName	*InterPart Relationship*	xsd: string
	owl:DatatypeProperty hasDescription	*InterPart Relationship*	xsd: string
	owl:DatatypeProperty hasValue	*InterPart Relationship*	xsd: string
	owl:ObjectProperty hasExeType	*InterPart Relationship*	*ExeType*
At sub-concept level for *AdjacentTo*	owl:ObjectProperty hasMcadPart	*AdjacentTo*	*MCAD_ Part*
	owl:ObjectProperty hasEcadPart	*AdjacentTo*	*ECAD_ Part*

The Protégé toolkit [17] with an OWL plug-in is used to develop the domain ontologies in this research. These ontologies explicitly specify the formal and intended semantics for the identified domain concepts, attributes of concepts, constraints on attributes, and relationships between concepts, etc. The concepts are mainly used in naming and searching the attribute identifiers of the design annotation expressions defined in Section 3.1. As ontologies represent a kind of semantic consensus of the concept definitions, the intended meanings of attribute values in design annotations can therefore be consistently interpreted and communicated among collaborating agents. On the other hand, the design ontologies also enable reasoning, query, and mapping over ontological definitions and relationships of domain concepts across heterogeneous design applications, so that the semantics of the design annotations can be shared among them.

3.3 Creation of semantic annotations

Semantic annotations are generated by instantiating the design annotation expressions in Section 3.1 with the information extracted from the existing design resources, from the relevant external applications, or from user's inputs. The meanings of the information content in the instantiated annotations are specified by the design ontologies in Section 3.2.

There are two types of instantiations to create annotation from annotation expressions and the given digital engineering resources:

- To explicate the semantics of the resource data based on the concept definitions in design ontologies; and/or
- To enrich the information meaning and information content from the resource with additional design semantics from other resources or from users.

They are handled differently. If the design information models, database schemas of the existing design resources are unambiguous and semantically complete to fulfill the identified annotation needs, then the instantiation of annotation expressions will be a process to transfer the meaning of the same information in different representations in order to make the meaning more explicit. However in engineering design reality, there always are differences in what is required in an annotation expression and what can be offered in a given design resource. Some information elements defined in the annotation expression and needed by design collaborators do not have semantic equivalences in the existing resource. This would usually require humans and external systems to supply additional semantics to enrich the data meanings of the existing resource. Semantics enrichment of design information for consumer products is covered in one of our other projects. What is elaborated here is how to explicate the design semantics in the annotation instantiation process.

Explicating the data meanings of the design resources can be accomplished by finding the most appropriate semantic information from a design resource for an attribute in an annotation expression. Design ontologies are used in this matching process to regulate the interpretation of the meaning of the attribute through *Attribute.Identifier* in Expression (1). The following lists two possible situations arising from the matching process. The other two possibilities for incomplete matching situations, which require semantics enrichment, are not covered here.

- There exists an exact match of meanings between an information element in the resource and an attribute in the annotation expression. The information element is therefore assigned to the attribute as its value.
- An information element in the resource is an equivalent value of an attribute in the annotation expression. The equivalence here refers to the same information content represented in different formats, units, etc.

In these situations above, the instantiation of annotation expressions can be an automatic process. Once the mapping rules are defined, software tools will support the auto-matching process. One of our early projects has developed such a tool [18] for explicating and extracting semantic information from collaborative product design (CDP) process models. Fig. 2 shows an interface of the tool used in explicating semantics of the CDP process data (in XML) to instantiate process annotations.

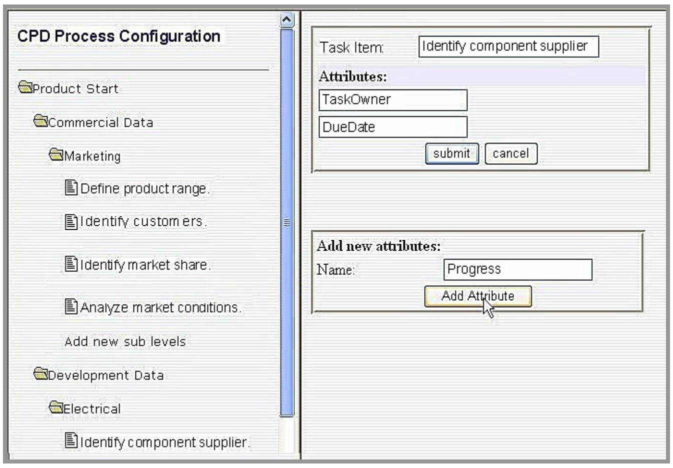

Fig. 2 Explicating the process data semantics

The semantic structure of these process annotations is defined by an annotation expression below. Examples of the instantiated process annotations from this expression are given in Table 2.

<PropertyName(String: Unique), Description(String), Value(String), Unit(String)>

Table 2 Examples of process annotations

PropertyName (Unique)	Description	Exemplary Value	Unit
TaskOwner	Responsible person of a task.	Jack Lee	
DueDate	Due date of a task.	15 April 2008	
Progress	Completion % of a task.	100	percent

The property names in Table 2 are retrieved from the controlled vocabulary library with their meanings and relations specified in domain ontology in Section 3.2. As such, the semantics of the process annotations can be meaningfully communicated among humans. To make them also interpretable and reusable in computer applications, the semantic annotations should be represented in some neutral, machine-readable formats as well.

3.4 OWL/RDF encoding of semantic annotations

The design annotations instantiated through the method above are readable and understandable to humans. They can also be embedded in CAD models to make them directly accessible and sharable among compatible CAD systems. To leverage machine-readability of the semantic annotations for both CAD and non-CAD applications, neutral encoding in an application-independent, platform-independent and machine-interpretable language is necessary. In the current research, we select OWL/RDF to encode the semantic annotations, because our design ontologies are in OWL which is also compatible to RDF.

OWL/RDF encoding marks up the semantics in design annotations according to the domain ontologies developed early. Through the encoding process, interpretive information elements in the annotations are organized in an XML-alike structure with semantic tags. These tags use commonly-agreed domain terminologies from the vocabulary library. Each tag has been semantically-specified by an OWL concept class in the design ontology. Thus the intended meanings of the information elements of the annotations can be exactly captured in the resultant neutral representations. Take the inter-part relationship annotation “Enclosure” for a *Base* part as an example. According to the OWL ontological definition for the *Enclosure* concept, the *Enclosure* relationship between a mechanical part “Base-P” and an electronic component “Multi-Pulse Laser Driver” can be encoded as follows.

```
<ECAD rdf:about="#MultiPulseLaserDriver">
  <rdf:type rdf:resource="&owl;Thing"/>
</ECAD>
<MCAD rdf:about="#Base-P">
  <rdf:type rdf:resource="&owl;Thing"/>
</MCAD>

<Enclosure rdf:ID="EnclosureAnnotation">
    <hasName rdf:datatype="&xsd;string">Enclosure_Base-P
    </hasName>
    <hasDescription rdf:datatype="&xsd;string">Multi-pulse laser
        driver is enclosed in Base-P of the optical pickup unit
    </hasDescription>
    <hasValue rdf:datatype="&xsd;string">enclosing</hasValue>
    <hasExeType rdf:resource="#java"/>
    <hasEnclosedPart rdf:resource="#MultiPulseLaserDriver"/>
    <hasEnclosingPart rdf:resource="#Base-P"/>
</Enclosure>
```

Fig. 3 OWL/RDF encoding for an *Enclosure* annotation

With the neutral representation in Fig. 3, the CAD and non-CAD applications can understand the meanings of any tags in the annotation such as *Enclosure*, *hasName*, *hasEnclosedPart*, etc. That is because they interpret the tags following the same set of semantic definitions in the OWL ontology. Thus, different disciplines can reuse the annotation content in a semantically consistent way. This helps overcome the terminological ambiguity in utilization of design annotations. The other potential usages of this exemplary annotation are detailed in one of our use scenarios in Section 4.1.

4. USE SCENARIOS

The following use scenarios in collaborative consumer product design will elaborate how semantic annotations could help achieve meaningful communications of design intent across engineering silos. Multiple CAD and non-CAD teams are involved in the two scenarios below.

4.1 Design intent communication across disciplines

The design communication process for a mechatronic device is considered in this scenario. Among others, the meanings of CAD-specific terminologies need to be annotated to support design intent communication between the involved CAD and qualification test teams.

By attaching semantic annotations to the CAD-specific terminologies and parameters used in CAD drawings and technical documents of the mechatronic device, the meanings of the terminologies are made explicit. The annotations also provide a reference to semantic definitions of the terminologies in design ontologies. Therefore, the domain terminologies can be searched and reasoned based on their ontological concepts to reveal the unknown relations and knowledge behind the CAD terminologies and parameters. This helps non-CAD disciplines build mutual understanding of designers' intentions. For example, in order to explain the meaning of the term "Enclosure" in the context of an inter-part relationship between an electronic CAD (ECAD) component *Multi-Pulse Laser Driver* and a mechanical CAD (MCAD) part *Base-P*, we can attach an enclosure annotation (as shown in Fig. 3) to a test plan of the mechatronic device. The annotation in Fig. 3 indicates that:

- The enclosure annotation is of type *Enclosure*, which means that the referred annotation is an instance of the ontological concept *Enclosure* as defined in Fig. 1;
- The *hasDescription* attribute value of the annotation provides textual description for human understanding of the annotation content;
- The *hasValue* and *hasExeType* attribute values together provide formal (machine-readable) definitions of the annotation content;
- The *hasEnclosedPart* and *hasEnclosingPart* attribute values specify the relationships between this annotation and the two involved parts: *MultiPulseLaserDriver* and *Base-P*; and
- The *MultiPulseLaserDriver* and *Base-P* are instances of the *ECAD* and *MCAD* ontological concept classes, respectively. This is derived from a formal reasoning over the *ECAD* and *MCAD* concept class definitions. The reasoning results are included in Fig. 3.

Assume that the qualification test team has a task: "to retrieve all the ECAD components of the mechatronic device, which are enclosed in the MCAD part *Base-P*, for reliability testing". To conduct the task, the test team needs to retrieve the required inter-part relations between *Base-P* and all its enclosed ECAD components. However, it will not be practical for the test team to use MCAD and ECAD packages to get these design relations. Using the *Enclosure* annotation attached with the test plan, the team and its testing system can access these inter-part relations. The testing system understands the intended meanings of the specialized CAD terminologies (e.g. *Enclosure*, *hasEnclosedPart*, etc) in the annotation, because the testing system and the annotation follow the same semantic agreements of the design ontologies, also because the annotation contents are expressed in the formal and neutral format. Thus, the designers' intention for testing only the ECAD components enclosed in *Base-P* is communicated to the testing team precisely.

4.2 Multi-CAD semantics sharing

This section explores how semantic annotations can support sharing of unambiguous multi-CAD semantics in the collaborative design process. An optical pickup sub-system (OPS) of DVD recorders is used as an example in the following scenario.

The multidisciplinary OPS design process starts with modeling of the initial assembly design for OPS geometry, board outline, keep-out regions, inter-part relationships/ constraints, assembly relations, etc. The assembly design information is shared by multiple design disciplines for them to do detailed designs on:

- MCAD modeling for part geometry, the dimension and location of mechanical fixtures, 3D obstacles, and enclosures, etc;
- ECAD modeling to decide the type, size and form of components, to place components on the board and to route traces based on the geometry restrictions given in the assembly design, followed by electrical circuit and component simulation; and
- Optical CAD design to select off-the-shelf components, to design light path geometries, and to analyze the optical sub-system to meet the specifications and constraints set in the assembly design.

It is identified that semantic annotations should be created to explicate and enrich the meaning and content of the OPS assembly drawings (created from AutoCAD system). The purpose is for better sharability of the critical assembly information among the involved disciplines, especially for those CAD systems that are not compatible with AutoCAD. Three types of semantic annotations are created for the OPS assembly drawings: CAD manipulations; inter-part relationships; and representation maps. Among them, an inter-part relationship for a behavioral annotation *AdjacentTo* is detailed in this section. The *AdjacentTo* annotation is intended to communicate the functional semantics of a design service across MCAD and ECAD systems. Fig. 4 shows the basic idea for multi-CAD semantics sharing in this scenario, which is followed by detailed discussion on the roles of the annotation in support of semantics sharing across multi-CAD platforms.

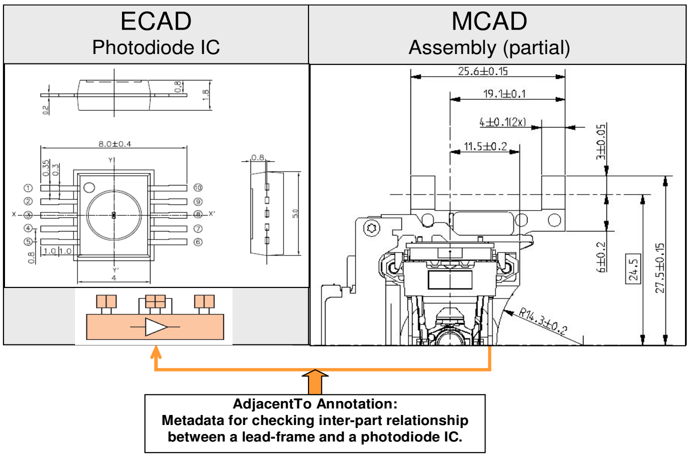

Fig. 4 Semantics sharing across multiple CAD platforms

The ontological definition of the *AdjacentTo* concept (refer to Table 1 for details) is instantiated to generate an OWL/RDF annotation in Fig. 5. The OWL/RDF code in Fig. 5 annotates a design service to check the *AdjacentTo* relationship between an electronic component design for *PhotodiodeIC* and a mechanical part design for *LeadFrame*.

```
<AdjacentTo rdf:ID="AdjacentTo_Annotation">
  <hasName rdf:datatype="&xsd;string">AdjacentTo_LeadFrame
  </hasName>
  <hasDescription rdf:datatype="&xsd;string">MCAD part
      Lead_Frame and ECAD component Photodiode_IC are
      adjacent to each other.
  </hasDescription>
  <hasValue rdf:datatype="&xsd;string">touchChecking</hasValue>
  <hasExeType rdf:resource="#cc"/>
  <hasMcadPart rdf:resource="#LeadFrame"/>
  <hasEcadPart rdf:resource="#PhotodiodeIC"/>
</AdjacentTo>
```

Fig. 5 Behavioral annotation for *AdjacentTo* relationship

The *AdjacentTo* annotation is an instance of *AdjacentTo* concept class. Same as the *Enclosure* concept in Section 4.1, the ontological concept *AdjacentTo* is also a sub-type of *InterPartRelationship* (refer to Fig. 1). Both inherit the same set of attributes from their super-concept class of *InterPartRelationship*. As such, the semantic structure of the *AdjacentTo* annotation in Fig. 5 is similar to that of the *Enclosure* annotation in Fig. 3.

During detailed design, when the ECAD system receives the *AdjacentTo* annotation attached to the OPS assembly specifications, its add-on tool will interpret the meaning of the annotation according to the semantic definitions specified in the *InterPartRelationship* concept class and the *AdjacentTo* sub-concept class in an ontology. By understanding the semantics of the *AdjacentTo* annotation, then invoking the design service designated in the annotation, the ECAD system will check and possibly adjust the design parameters of *PhotodiodeIC* to meet the requirements of the OPS assembly. On the other hand, the MCAD system will also perform the similar inter-part relationship checking to the mechanically-designed *LeadFrame* part. In this way, the functional semantics for checking of the *AdjacentTo* relationship are unambiguously shared in MCAD and ECAD to achieve the overall goal of the OPS design.

5. CONCLUSION

A new approach has been introduced for capturing, formalizing and annotating design semantics to support seamless collaboration in multidisciplinary consumer product design. The approach takes advantage of engineering design annotation methods and Semantic Web technologies to achieve ontology-based semantics sharing. The proposed approach is fundamentally different from the traditional design annotation methods as it specifies the meaning of annotations with design ontologies and it uses OWL/RDF to encode annotation semantics for meaningful communication across disciplines. The multidisciplinary design teams and systems would benefit from these features for explicit and formal sharing of design semantics of heterogeneous engineering resources in neutral annotation formats to support seamless collaborations in consumer product development.

Our future work will be focused on annotation ontology mapping across organizational boundaries to support product knowledge sharing.

REFERENCE

1 Lee J (1997) Design rationale systems: understanding the issues. IEEE Expert 12(3):78-85

2 Gruber TR, Russell DM (1995) Generative design rationale: beyond the record and replay paradigm. In: Moran T, Carroll JH (eds) Design rationale: concepts, techniques, and use. Lawrence Erlbaum Associates, pp 323-349

3 Aubry S, Okawa S, Lenne D, Thouvenin I (2005) Knowledge integration in annotation-based collaborative virtual environments. In: Proceedings interaction 2005, Tokyo, Japan

4 LaViola JJ, Holden LS, Forsberg AS, Bhuphaibool D, Zeleznik RC (1998) Collaborative conceptual modeling using the SKETCH framework. In: Proceedings of the IASTED international conference on computer graphics and imaging, 154-158

5 Olive J, Thouvenin I, Lenne D (2005) A new knowledge visualization and manipulation metaphor for collaborative and immersive environment. In: Proceedings of the 2nd international conference on enactive interfaces, Genoa, Italy

6 Do EYL (2001) VR Sketchpad: create instant 3D worlds by sketching on a transparent window. In: Proceedings of the 9th international conference on computer-aided architectural design, Eindhoven, The Netherlands, pp 161-172

7 Beckett D (2004) RDF/XML syntax specification, W3C recommendation, http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/

8 Dean M, Schreiber G (2004) OWL Web ontology language reference, W3C recommendation, http://www.w3.org/TR/2004/REC-owl-ref-20040210/

9 Nagarajan M (2006) Semantic annotations in web services. In: Cardoso J, Sheth AP (eds) Semantic web services, processes and applications, Springer, pp35-61

10 Uren V, Cimiano P, Iria J, Handschuh S, Vargas-Vera M, Motta E, Ciravegna F (2006) Semantic annotation for knowledge management: requirements and a survey of the state of the art. J of web semantics: science, services and agents on the World Wide Web 4: 14–28

11 Reeve L, Han H (2005) Survey of semantic annotation platforms. In: Proceedings of the ACM symposium on applied computing, New Mexico, USA

12 Kitamura Y, Washio N, Koji Y, Mizoguchi R (2006) Towards ontologies of functionality and semantic annotation for technical knowledge management, In: New frontiers in artificial intelligence: joint JSAI 2005 workshop post-proceedings, Springer-Verlag Berlin / Heidelberg, pp 17-28

13 Lin Y, Strasunskas D, Hakkarainen S, Krogstie J, Sølvberg A (2006) Semantic annotation framework to manage semantic heterogeneity of process models. In: Proceedings of the 18th conference on advanced information systems engineering, Springer-Verlag Berlin Heidelberg, pp 433-446

14 SEVENPRO project (FP6-027473). (2006) SEVENPRO annual public report 2006. http://www.sevenpro.org/

15 ATHENA project (IST-507849). (2004) ATHENA key results & ATHENA performance indicators. ATHENA project deliverables D.C1.1/D.C2.1 & D.C2.1, http://www.athena-ip.org/

16 Boudjlida N, Boudjlida N, Dong C, Baïna S, Panetto H, Krogstie J, Hahn A, Hausmann K, Tomás JV, Poler R, Abián MÁ, Núñez MJ, Zouggar , Diamantini C, Tinella S (2006) A practical experiment on semantic enrichment of enterprise models in a homogeneous environment, INTEROP-NoE project deliverable DTG4.1

17 Stanford University (2007) Protégé release 3.3: Protégé-OWL editor, http://protege.stanford.edu/

18 Yang QZ, Zhang Y (2007) Semantic interoperability to support collaborative product development. In: Li WD, Ong SK, Nee AYC, McMahon C (eds) Collaborative product design and manufacturing methodologies and applications. Springer, London, pp 255-286

19 Shipman III FM, McCall RJ (1996) Integrating different perspectives on design rationale: supporting the emergence of design rationale from design communication. Technical report CSDL 96-001, Texas A&M University, USA

20 Gruber TR, Tenenbaum JM, Weber JC (1992) Toward a knowledge medium for collaborative product development. In: Gero JS (ed) Artificial intelligence in design: proceedings of the 2nd international conference on artificial intelligence in design. Kluwer Academic Publishers, Boston

21 Welty C, Smith MK, McGuinness DL (2004) OWL Web Ontology Language Guide, W3C recommendation, http://www.w3.org/TR/2004/REC-owl-guide-20040210/

