An Ontology-Driven Hierarchical Modeling Method for Multi-Disciplinary Collaborative Design

Yongping Hao, Chunyan Wang*, Pengfei Zeng, and Xiaolei Xu
R&D Center of CAD/CAM Technology, Shenyang Ligong University, Shenyang 110159, China

A hierarchical modeling method is proposed for multi-disciplinary collaborative design which is driven by ontology. This method is focused on defining five states ontologies and nine description layers so as to describe design processes in a hierarchical and standardized level. The nine layers are instance (I) layer, concrete (C) layer, concept-model (CM) layer, meta-model (MM) layer, design-process (DP) layer, data (D) layer, data-integration (DI) layer, model-integration (MI) layer and design-completed (DC) layer, furthermore, specific contents of these layers are illustrated in detail. Through description using a series of related and hierarchical layers, design processes of multi-disciplinary collaborative achieve flexibility in structures. Finally, state ontologies, which are defined to store description layers, can also facilitate management and implementation of product design processes by clearly reflecting process of multi-disciplinary collaborative design that is driven by ontology.

Keywords: Multi-Disciplinary Collaborative Design, Ontology, Hierarchical Modeling, Description Layers.

1. INTRODUCTION

Multi-disciplinary, such as structure discipline, aerodynamic discipline, control discipline, ballistic discipline, promote discipline, statistical discipline, quality discipline, environmental discipline, etc. are needed in products collaborative design, especially for complex products. However, each discipline has its own mathematical foundation and organization form. In addition, different disciplinary experts have their own backgrounds and may have different perspectives on the same product. Problems of semantic heterogeneity and syntax heterogeneity will also exist. Hence, it is necessary to standardize the design process in a hierarchical level with the purpose of achieving multi-disciplinary collaborative design and avoiding restrictions caused by multi-disciplinary.

To solve issues of standardizing design processes and avoiding heterogeneous problems caused by multi-disciplinary, a hierarchical modeling method, in which knowledge is conducive to use and reuse, is proposed for multi-disciplinary collaborative design. In this method, nine description layers are defined so as to describe design processes in a hierarchical and standardized level, and specific contents of these layers are illustrated in detail. Through description using a series of related and hierarchical layers, design processes of multi-disciplinary collaborative achieve flexibility in structures. Additionally, state ontologies which are used to store description layers can also facilitate management and implementation of product design processes by clearly reflecting the level of the design process. Modeling method in this paper is driven by ontology, which are a philosophical concept and a branch of metaphysics that deals with the nature of being. Researches on ontology have recently been increasing for its advantages in semantic and knowledge levels, ontology can be seen as a well-defined description for concepts and can also be seen as an objective description for both concepts and relationships. Hence, understanding bottleneck among different disciplinary experts will be resolved by ontology method.

2. HIERARCHICAL MODELING AND DESCRIPTION LAYERS

2.1. Hierarchical Modeling

Hierarchical modeling referred in this paper is a hierarchical description for collaborative design processes based on multi-disciplinary which is driven by ontology. Nine description layers are defined to describe structures and relationships of data and information elements within process of collaborative design, and then to achieve hierarchical modeling.

Hierarchical modeling method is mainly to divide a design process into nine description layers and five state ontologies having one or more description layers. The state ontologies show different states of a design process and reflect different stages of product development. This method is focused on defining five states ontologies and nine description layers so as to describe design processes in a hierarchical and standardized level.
of a design. And we have defined five kinds of state ontologies
which are project ontology, domain ontology, task ontology, data
ontology and product ontology. Figure 1 shows the five kinds of
state ontologies, also can be seen form this figure:
(1) Project ontology is formed with basis of instance (I) layer
and concrete (C) layer.
(2) Domain ontology is formed with basis of concept-model
(CM) layer.
(3) Task ontology is formed with basis of meta-model (MM)
layer.
(4) Data ontology is formed with basis of design-process (DP)
layer, data (D) layer and data-integration (DI) layer.
(5) Product ontology is formed with basis of model-integration
(MI) layer and design-completed (DC) layer.

And these layers are defined in the following, and all of these
layers are existed in the same order shown in Figure 1.

2.2. Description Layers

Nine description layers involved in the hierarchical modeling
method consist of:
(1) Instance (I) layer: Definition of instance layer is needed to
meet the diversity of design products, and an instance represents
one kind of product. In other words, if a product to be designed
is water-pumping then the instance layer should be assigned to
water-pumping, without exception—if a designed product is mis-
siles then the instance layer should be assigned to missile.
(2) Concrete (C) layer: Concrete layer reflects preparatory work
to be done before a design process is beginning. Works of
requirement analysis, target index and data preparation are
referred in this layer, also design experience should be contained
which supplies certain basis for task decomposition.

(3) Concept-model (CM) layer: Concept-model layer derives
from results of the concrete layer and depends on its processing
data. In this way, parameters and design specifications for
the product to be designed, in the concept-model layer, should
be described as detailed as possible, and it is the basis for
establishing domain ontology. For example, Figure 2 shows a
parameter with sub-parameters of speed and position, and then
sub-parameters of speed and position should contain param-
ters of their own. And, Figure 2 is produced by using protégé
(http://protege.stanford.edu/), which is a set of tools designed
to automate the process of building domain-specific knowledge
acquisition. We use the edition of protégé 4.0.2.
(4) Meta-model (MM) Layer: Meta-model layer is formed by
concept-model decomposition which is based on meta-model.
Meta-models and corresponding tasks from decomposition are
stored in this layer. Orders of tasks are marked in such rules—
taking taskij for example, in which, i represents this task is from
which meta-model, and j represents order in its task.
(5) Design-process (DP) Layer: Design-process layer is a main
layer for a product design. In this paper, we set multi-disciplinary
as an entry point to process description in a product designing
process. Given that multi-disciplinary will be used to finish a
product's design, especially in complex product designing. For
example, if we want to design an electric-hydraulic pump for a
car, then disciplines of mechanical dynamics, hydraulic, multi-
body dynamics and electronic are used in its collaboration design
process. Different disciplines, are used to achieve single one task,
get together to form a discipline-combination. Supporting that
not only one discipline-combination are existing, the most suit-
able discipline-combination will be found for finishing one task
through intelligent search and human experience. Figure 3 illus-
trates flow chart of Taskij with getting its most suitable subject-
combination from many discipline-combinations.

As previously seen, we have turned products design into tasks
finishing which is fulfilled in multi-disciplinary collaborative
design through description layers. Now, we only need to com-
plete each task, rather than considering the whole design process.
(6) Data (D) layer: Data layer is defined for storing data gen-
erated by the design process. Data to be stored, including data
from different analysis software and from different disciplines,
has problems of hard to share and reuse. So as to overcome the
Fig. 3. Flow chart for Taski to get its most suitable discipline-combination.

problems, we have to resolve it by classifying the data according to types firstly. And three kinds of states which are compatible, structural incompatible and semantic incompatible are defined for a course of data’s share and reuse. The three data states are shown in Figure 4 by taking data Ds and Dt as an example, also the two data are to communicate, what’s more, Figure 4 illustrates corresponding mappings which will appear at where data incompactness, including structural incompatible and semantic incompatible. In Figure 4, mapping (δ1) and mapping (δ2) have differences in definitions, although they have the same target of making data compatible.

(1) Mapping (δ1): Mapping (δ1) is defined for data’s structural incompatible. In other words, mapping (δ1) is used to solve problems caused by data’s structural incompatible. In this paper, mapping (δ1) is a rule within data storing at a consistent form, hence we are supposed to use storage form of the tree which is consistent with ontology that is stored and edited by protégé 4.0.2.

(2) Mapping (δ2): With propose of resolving problem of data’s semantic incompatible, mapping (δ2) is defined. As prevision seen, data’s semantic incompatible can mainly avoided in ontology modeling method which has its own advantage of knowledge sharing, especially semantic sharing. In our definitions, data’s semantic incompatible has two aspects. The one is that different terms may be used to denote the same entity, and the other is that the same term is being applied to different concepts. Modeling method driven by ontology can avoid the two aspects of data’s semantic incompatible with its modeling advantages, and the solutions seen below, which are in an ontology modeling environment:

- Solution to avoid that different terms may be used to denote the same entity: This solution is produced in the time of ontology (the ontology is used to store data from design process) building. Hardly not only one term are referred to single entities, when protégé will prompt you to define them as equivalent relationships. In other words, protégé will let you make terms equivalent to fight for semantic incompatible.

- Solution to avoid that the same term is being applied to different concepts: In the protégé, it will prompt you that this term

Fig. 4. Tasking Ds and Dt as an example to show three kinds of data states in communication and mappings among them.
already exists when the same term has appeared not only once. You can replace another term or add names of class or subclass as a prefix to distinguish them.

(7) Data-integration (DI) layer: Data-integration layer will appear when there are data to be integrated. The data can be formed between different disciplines, also can be got among different functional software within the same discipline. Data integration is a core issue of building a sharing model for multi-disciplinary collaborative design, but also to measure whether the modeling method is successful or not.

(8) Model-integration (MI) layer: Model-integration layer occurred after data integrations. When data integration has completed, it indicates that all design tasks are completed. In order to get the whole product, all design tasks should be integrated together, we call the process which is used for all tasks’ integration as a model-integration layer. The following problems always occur during model integrated process.

- Mismatch of model concepts: In this mismatch, concepts from different models are not matching each other. So, we should define correspondence relationships to make sure that every concept can be the same thing in all models.
- Mismatch of model properties: Model properties are used to describe their concepts, and they must not change in a product design. If one concept has different properties in models to be integrated, there will be a mismatch of model properties.
- Mismatch of model versions: In this paper, model versions are the versions of disciplinary analysis tools. One tool should be keeping the same version run through the whole product design. For example, mismatch of model versions will occur when both CATIA V4 and CATIA V5 are used in one product design.

(9) Design-completed (DC) layer: Design-completed layer is a sign of product design’s completion.

3. HIERARCHICAL MODELING PROCESSES

In this section, steps for hierarchical modeling driven by ontology are summarized as follows:

Step 1: Firstly, an instance layer is formed by assigning a specific product to be designed which is shown in Figure 5.

Step 2: Requirement analysis, target index and data preparation for designed product are done so as to constitute a concrete layer (shown in Fig. 6), such as works of technical indicators and description parameters.

Step 3: With the completes of definition for the instance layer in step 1 and the concrete layer in Step 2, product ontology is formed based on the two layers, and it is shown in Figure 7.

Step 4: In this step, technical indicators and description parameters for the designed product are depicted as detail as possible. And a concept-model layer which is based to become domain ontology is formed. The course for domain ontology forming can be seen in Figure 8.

Step 5: We should decompose the concept model in concept-model layer into finite numbers of meta-models in order to facilitate realization and management of design processes, and then each meta-model is divided into several design tasks using labels of takeij, in which the i is the number of meta model, and its data type is integer, in addition, j is the number of a task in all tasks, and it uses capital letters of A, B, C, ..., et al. In this way, a meta-model layer and its corresponding task ontology are formed. Figure 9 shows an example of a meta-model layer with three meta-models and fewer tasks present.

Step 6: We have focused on multi-disciplinary as an entry point of the design modeling method in this paper, that is to say, we are supposed to reflect disciplines in design products. As is known to all, design knowledge of one or more disciplines are needed
for each of design task. Further more, each discipline has its own disciplinary analysis tools, and all of domains experts have knowledge background belonging to their own fields. So, design disciplines, disciplinary analysis tools, domains experts, designers and other resources constitute a design-process layer. Several commonly used disciplinary and their analysis tools are listed in Figure 10.

---

**Fig. 10.** Several commonly used disciplines and their analysis tools.

---

**Step 7:** Enumeration of design data’s formats. In Figure 11, three kinds of forms are listed.

---

**Fig. 11.** Three kinds of data’s formats.

---

**Step 8:** In order to complete products design, data from data layer needs to be integrated in data-integration layer. Completing for design tasks is a course for data integration. Three kinds of states for data sharing and reusing, which are compatible, structural incompatible and semantic incompatible, are defined when data is connecting. Figure 12 shows these states and mappings.

---

**Fig. 12.** States and mappings for data and a data-integration layer.

---

**Step 9:** The design-process layer, the data layer and the data-integration layer are used to form data ontology, and this is shown in Figure 13.

---

**Step 10:** As the data has integrated, what should do next is to make all meta-models integrated at that time, and we call this level as a model-integration layer.

---

**Step 11:** A design-completed layer occurring to end this product design.

---

**Step 12:** Finally, by using layers of model-integration and design-completed, product ontology is formed to describe designed products by using ontology, we use protégé 4.0.2, in addition, human factors are needed, to finish this step.
4. SUMMARY

In this paper, an ontology-driven hierarchical modeling method is proposed to make design processes with a hierarchical description so as to achieve design management more convenient and to make design process more standardized. The general idea of this method is described by defining five kinds of state ontologies with nine description layers. Specific contents of these nine description layers—instance layer, concrete layer, concept-model layer, meta-model layer, design-process layer, data layer, data-integration layer, model-integration and design-completed layer are elaborated in detail. Given that problems of semantic heterogeneity for multi-disciplinary collaborate design are still existing, furthermore, experts of different disciplines have different opinion on the same product to be designed for they have knowledge of their own fields, and each of them may have different design views. With the purpose of overcoming these problems, hierarchical modeling method is put forward driven by ontology which has natural advantage of solving the semantic heterogeneity.

In this modeling process, heterogeneities of different terms may be used to denote the same entity and that the same term is being applied to different concepts can be avoided and solved by using ontology. It can be proved that the hierarchical modeling method reflects flexibility for design process management. Problems from a design process should be boiled down to a certain layer, instead of changing the whole modeling process. Using a series of related and hierarchical description and using ontology modeling with standardization of design processes, design knowledge becomes easier to share and reuse for both the present designers and the future ones.

There should be more applying examples using this hierarchical modeling approach in the future work. In addition, other detailed layers will be raised in the model as a supplementary of this modeling method.

Acknowledgments: This paper is supported by the National High-Tech. R&D Program of China (863) (No. 2009AA04Z167) and the National Natural Science Foundation of China (No.50975183).

References and Notes