4

Classification of Knowledge for Generating Engineering Models

-A Case Study of Model Generation in Finite Element Analysis-

Takayuki Sekiya, Akira Tsumaya, and Tetsuo Tomiyama
Research into Artifacts, Center for Engineering (RACE), The University of Tokyo, Japan

Key words: Model Generation, Engineering Model, Ontology, Knowledge Intensive Engineering

Abstract: During design, a designer uses various computational tools, such as a geometric modeling system, analysis tools, and databases. To support these design processes, a system that can integrate such computational tools and support him/her to use the tools and to build, operate and modify models is required. For this purpose, we have been developing the Knowledge Intensive Engineering Framework (KIEF) system. This paper describes the structure of knowledge model used in a modeling process of engineering models. We analyzed engineering knowledge used for Finite Element Analysis (FEA), categorized it, and finally built an environment to support FEA on KIEF.

1. INTRODUCTION

Recent progress of both software and hardware technologies made computational tools, such as geometric modeling systems and analysis tools, popular. Although we have already various convenient tools, a designer sometimes realizes that he/she does not have powerful ways to transfer data from one software to another and that he/she has to input a similar data again and again.

S. Finger et al. (eds.), *Knowledge Intensive Computer Aided Design*

To solve this problem, integrated CAE (Computer Aided Engineering) systems were developed. Most commercial CAE systems allow the designer to transfer CAD data to a numerical analysis tool such as an FEA system, based on product modeling technologies and STEP (Fowler, 1995). They support designers' engineering activities including design and analysis. Typically automotive and aircraft industries heavily use such systems for their engineering activities.

However, model construction is a quite intellectual process and requires the designer's appropriate judgement, including questions like which part of a product must be modeled and which kind of conditions must be applied to the model. This signifies that to develop a system which integrates various design tools, and that supports only data exchange is not enough.

When building a model, the designer uses a wide variety of engineering design knowledge, from common sense knowledge about the physical world to domain specific knowledge about how to use tools. The Knowledge Intensive Engineering Framework (KIEF) (Tomiyama, 1994b, Tomiyama, 1995) is our attempt to provide designers with such modeling knowledge in an integrated manner.

In this paper, we describe the structure of engineering knowledge required for model building. First, we analyze modeling processes, and extract modeling knowledge used in these processes. Next, we categorize it and propose its knowledge structure represented on KIEF. Finally, we illustrate an example to represent modeling knowledge.

2. THE MODELING KNOWLEDGE OF THE KNOWLEDGE INTENSIVE ENGINEERING FRAMEWORK

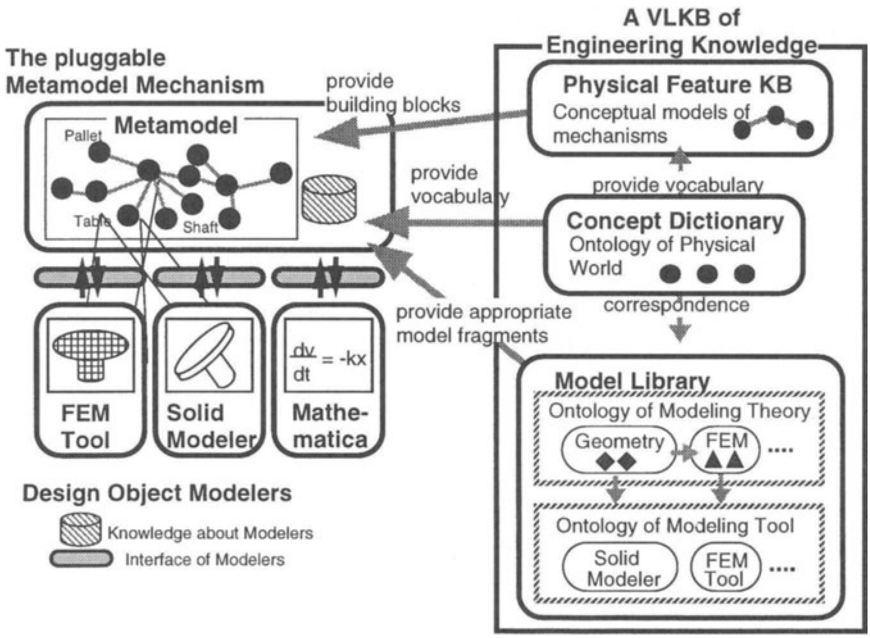
We have been developing KIEF that enables the designer to integrate existing design tools, and supports him/her in such processes as building, operation, and modification of models. There are similar approaches to our study and some of them aim at full automation of modeling.

2.1 Modeling Automation Approaches

One of our research interests is to represent the designer's thought process in model building. Research in automated modeling in AI is based on the notion that the computer solves a modeling problem. In automated modeling, there are several approaches such as model composition and model selection.

The model composition approaches construct a model by combining predefined model fragments. A designer specifies a scenario description and some queries to be answered by the system. A scenario description describes the object to be modeled in terms of components and their connections. The model construction system then uses a library of model fragments to construct a model. For instance, Falkenhainer's Compositional Modeling (Falkenhainer, 1991) and Nayak's approach (Nayak, 1996) fall into this approach.

The model selection approaches prepare several types of models beforehand for a physical domain, and each model has conditions to be satisfied. The designer specifies conditions to hold for the model. The system searches an appropriate model in the library of models. For instance, Addanki's Graphs of Models (Addanki, 1991) is a typical one.



2.2 Knowledge Intensive Engineering Framework

KIEF is a system that integrates existing design tools and supports the designer's activity on the tools, such as model building, model-based reasoning, and model validation. Figure 1 depicts the architecture of KIEF.

The main features of KIEF are the **pluggable metamodel mechanism** (Yoshioka, 1993) that is a mechanism for integrating design tools and the

Very Large-scale Knowledge Base (VLKB) (Ishii, 1995) which supports the metamodel mechanism by supplying primitive knowledge about the physical world.

The metamodel mechanism has symbolic representation of concepts about physical phenomena and mechanical components. A **metamodel** of a design object is represented as a network of relationships among concepts that appear in aspect models. Types of relationship include causal dependency among physical phenomena, arrangements of components, and quantitative relationships. These concepts and relationships constitute the ontology of KIEF. An existing external modeler however, usually deals with an aspect model. Therefore it is desirable that we could plug such existing modelers into KIEF. The **pluggable metamodel mechanism** allows easily plugging in external modelers into KIEF (Yoshioka, 1993). Each external modeler can access data defined in other modelers through the pluggable metamodel mechanism.

VLKB for KIEF supplies fundamental knowledge about the physical world. VLKB consists of the three primary knowledge bases, **concept dictionary, model library**, and **physical feature knowledge base**. The concept dictionary contains the fundamental and general ontology of KIEF, such as physical concepts and relationships among concepts. The ontology is used for building a metamodel and representing engineering knowledge. The model library stores ontologies specific to external modelers, such as model fragments of external modelers and correspondence between them and physical concepts the concept dictionary provides. The physical feature knowledge base contains **physical features** that are combinations among physical concepts. The physical features are the statements that describe physical situation, and the designer constructs a metamodel with physical features.

Roughly speaking, a model building process on KIEF consists of two steps. The first step is to extract a part of the conceptual network in the metamodel that is related to the tool that the designer wants to use, and to convert each concept to a tool-specific concept. We call the network of concepts extracted in the first step, an **aspect model**. The second step is to add quantitative information to an aspect model, and then to prepare data for further numerical analysis.

Through these modeling steps, the designer makes decisions for modeling, while the system just presents possibilities (such as tool-specific concepts) to him/her by automatically retrieving data from appropriate tools and calculates it after the designer specifies necessary data.

2.3 Our Approach

There is a gap between the AI approaches reviewed in Section 2.1 and what we aim at. These AI approaches are based on the idea of automating the entire model building process for a specific domain, and require the existence of a library which stores well-prepared knowledge. However, from an engineering point of view, there are too many design objects and too many ways to analyze design objects. This means that it is quite impossible to prepare a complete library. The designer does not necessarily expect the system to build a model fully automatically.

To integrate design tools on KIEF, the system has to tell the designer which tool can be used for a certain design object, for a certain goal, and in a certain situation. Therefore, what we first need is to understand how the designer builds a model and to systematically represent the designers' knowledge.

3. KNOWLEDGE REQUIRED FOR A MODELING

This section describes an analysis of model building knowledge based on observation of an actual modeling process.

3.1 The Modeling Process for Structural Analysis of a Turntable Mechanism

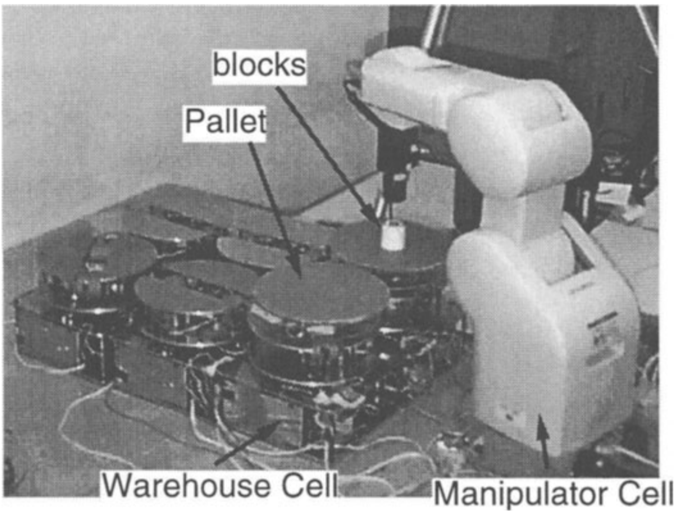


Figure 2: The Cellular Automatic Warehouse

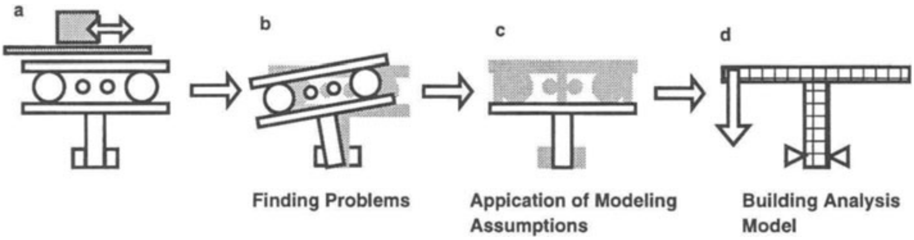


Figure 3: The Structural Analysis of a Cell

Our group has been developing a mechanical system called "Cellular Automatic Warehouse" which consists of homogeneous intellectual autonomous cells arranged in a two dimensional space (Sakao, 1996) (See Figure 2). Each cell has its own CPU and a turntable with rollers, and can transfer a pallet to neighboring cells.

We observed structural analysis of the turntable mechanism that is a part of a cell and identified information required for modeling. The whole modeling process from finding problems to building an FEM model is illustrated in Figure 3. First, we roughly classified the information required for this modeling process into engineering knowledge, a description of a design object itself, and reasoning results derived from them. The followings are the all information explicitly observed.

1. Engineering Knowledge
   (a) Phenomenon Related to Analysis
   - If a heavy object on another object is moving, then the load applied to the latter object will always be changing.
   - If an object is loaded, then it will deform.

   (b) For General Analysis
   - Simpler model is preferred.
   - If the stiffness of material is low, it is easy to bend.
   - Deformation can be detected by structural analysis.

   (c) Knowledge about FEM
   - If a twisting moment does not work on an object, it can be regarded as a shell.
   - (In many cases), minor details are negligible.
   - Simpler models, such as those consisting of simpler elements and fewer elements, can save analysis costs and time.
2. Descriptions of a Design Object
   (a) Functions, structures, and behavior of the turntable mechanism
   - The table is thin.
   - A shaft supports the table.
   - A heavy load on the table moves.
   - The cells are spread on a plane.

- The table rotates with a pallet.
- The machine delivers a pallet to neighboring cells.

(b) Goal of Analysis

- Estimate approximately how much the table will deform.

3. Reasoning Results

(a) Necessity of Analysis

- To check if the deformation of the table and the shaft prevent the machine from working as the designer expects.

(b) Reasoning Result for Structural Analysis

- The detailed structure of the table is neglected, because the goal of analysis is to estimate approximately how the table will deform and because simpler models save cost and time for analysis.

3.2 Findings in the Model Building Process

We analyzed the process of building an FEM model of the turntable mechanism and found out the following issues important.

3.2.1 Relationships between the Goal of Analysis and the Information of Design Object

Most AI approaches to automated modeling are appropriate for dealing with modeling process after a modeling goal is defined. In an engineering modeling process, a designer usually starts the modeling process with an informal problem statement (Pos, 1997). On the other hand, an experienced designer knows what kind of problems should be considered for design. He/she has knowledge about correspondences between design objects and such problems, and the knowledge about information of the design object, such as function, behavior, and structure. In case of the turntable design, the designer analyzes deformation of the table, because it works against one of the fundamental functions to deliver a pallet, or the structure of the turntable has little resistance against load. It will be helpful for less experienced designers to detect problems that occur to the design object.

3.2.2 Selecting an Analysis Method and Determining of Conditions Required for Analysis

During the design process, the designer repeats analysis of the same design object while he/she changes parameter values, according to the result of the analysis. So it is important to save cost and time for one analysis step as much as possible. The designer always has to select an appropriate analysis method and appropriate modeling conditions for a modeling goal.

4. CLASSIFICATION OF ENGINEERING KNOWLEDGE REQUIRED FOR ANALYSIS

In Section 3.1, we analyzed an analysis process from finding a problem to building an analysis model on KIEF. We classify the modeling process into the following four stages (see Table 1). In the following sections, we consider what the designer is doing and how to support him/her at each stage.

4.1 Design Stage

First, the designer specifies a domain of the modeling problem and a goal of modeling in this stage, and considers what problems will occur to a design object after that. Skilled designers can guess problems to be considered by looking at the design object, because they can compare the situation with their experiences and imagine how the design object works.

Table 1: Steps of Modeling Process

	Description	Model Representation for Reasoning	Knowledge
Design Stage	Choose Problem and Derive Unexpected Phenomenon	Part Relationship Network	Relationships between Function and Problem
Analysis Stage	Select Tools and Determine Conditions	Parameter Network	Related Parameters to Physical Phenomenon Model Modification Rules
Modeling Theory Stage	Map to Modeling Theory	Model Specific Representation	Theory specific Rules
Tool Stage	Convert Data for Modeling Tool	Parameters	Tool Specific Rules

Theoretically, if our knowledge is perfect, it is possible to derive every phenomenon that may occur to the design object. It is not realistic to do so, but the designer still somehow has to select relevant phenomena. Therefore, we will look for a mechanism to highlight only necessary phenomena.

As mentioned in Section 3.2.1, the goal of analysis has some connections with information of the design object. Therefore, in this research we build a knowledge base that contains correspondences between function, behavior, structure and problems that should be considered. The designer selects some problems to be solved after the system proposed problems to be considered with the knowledge base.

```
Procedure extract-related-rules(phenomenon, answer)
   /* extract reasoning rules related with the specified physical phenomenon
    phenomenon */
   phenomenon, conditionPP : physical phenomenon;
   collectionPP : collection of physical phenomenon;
   pf : rule of reasoning phenomenon;   pfCol, answer : collection of rule;
   RuleBase : all reasoning rules;
begin
   for every pf stored in RuleBase do
   begin
    collectionPP = get-all-phenomena-reasoned-out-by(pf);
    /* get all physical phenomena reasoned out by applying a rule pf */
    if (collectionPP includes phenomenon) and not (answer includes pf) then
    begin
      add pf to answer;
      for every conditionPP in (get-all-phenomena-to-reason-out(pf)) do
         /* get all physical phenomena required to apply a rule pf */
             extract-related-rules(conditionPP, answer);
    end;
   end;
end;
```

Figure 4: The Algorithm to Extract Related Rules

Next, the system reasons out physical phenomena that may occur to the design object, and that are relevant to the problems to be considered. Supposing that the system stores a large number of reasoning rules, a lot of phenomena, some of which may be fatal, and others of which may be trivial, are derived. They may prevent the designer from understanding important problems to be considered for design. Therefore we implemented an algorithm to extract rules related only to the phenomena which are specified beforehand as unrelevant phenomena that cause problems (see Figure 4). The procedure, "extract-related-rules," recursively searches rules used for reasoning out a specified phenomenon, "*phenomenon*." The designer moves to the next step, after the system proposes unexpected phenomena that occur to the model.

4.2 Analysis Stage

In this stage, the designer selects an analysis method and determines conditions required for analysis. In order to support the designer to select an appropriate analysis method, it is necessary to prepare knowledge about analysis methods themselves as follows.

- Inputs to analysis methods

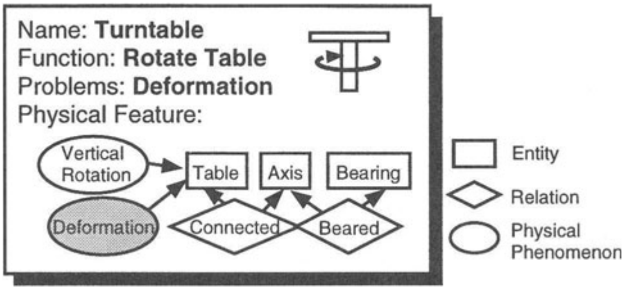


Figure 5: Knowledge about a Goad of Analysis

- Outputs from analysis methods
- Required conditions to execute the analysis

Determining the extent and conditions for analysis is as important as selecting an analysis method. For example, in case of FEM, it is seldom to generate mesh data from original CAD data without processing it. Detailed data is negligible because of Saint Venant's principle. Usually, the designer judges whether or not each geometric data can be regarded negligible when building an FEM model. It is also important to set constraints in the model. There might not be a general way to represent such designer's knowledge as rules.

However, for example, consider a case that the allowable maximum stress is given, and the goal of analysis is to confirm if applied stress never excesses the maximum value. However, the value of stress calculated by analysis will be larger than the true value even if the designer neglects the parts for reinforcement. Therefore, the true values will not excess the maximum value if the value with the simplified model does not excess it. In short, “The detailed data does not have to be paid attention to when neglecting it affects a simulation result and it falls into the safe side comparing the result with the goal of analysis.” We call this heuristic rule the reliable minimum one. In order to use such a kind of rules, we will build a parameter network representing that parameters are proportion to one another.

In our KIEF system, we have a qualitative reasoning tool to analyze a qualitative behavior of a design object. To build a model for this tool, the system provides knowledge about correspondence between physical phenomena reasoned out at a previous stage, and qualitative relationships among parameters that are related to the phenomena. In this research, we will use this type of knowledge to build a parameter network.

4.3 Modeling Theory Stage and Tool Stage

General concepts are converted into concepts specific to the modeling theory. There must be a domain specific theory and a domain specific ontology, and a domain specific way to support modeling. The ontology in the modeling theory stage is dependent on a modeling theory, but independent of a modeling tool. In case of FEM, the ontology of FEM includes such elements for FEM as beam, shell, constraints, and materials. These elements are general to any FEA system, but each system uses is own representation and data format for these elements.

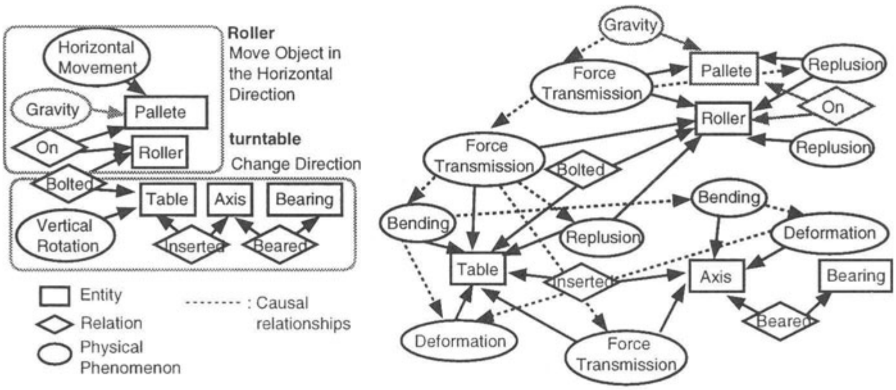
The primary work on the tool stage is to prepare the data required for executing the analysis process with a specific tool.

5. SYSTEM ARCHITECTURE AND AN EXAMPLE

This section presents an overview of the system architecture (see Figure 1) and discusses in detail various system components using an example.

5.1 System Overview

In this research, we add some knowledge bases and tools to our existing KIEF system to support model building based on the ideas described in the previous section. The following features are developed for this research.



a. A metamodel of the turntable mechanism, which represents the structure, the behavior (sub-graphs surrounded by rounded rectangles.)

b. The model after reasoning out physical phenomena which occurr to a turntable mechanism. (Some data are hidden for convenience to understand.)

Figure 6: A Metamodel of the Turntable Mechanism

- Knowledge base about a goal of analysis (not yet implemented)
  We build a knowledge base that contains correspondences between function, behavior, structure, and problems that should be considered. Figure 5 depicts a turntable mechanism to realize the function, "To rotate a table," and a deformation of the table to be considered.
- External tools (A solid modeler and an FEM solver)
  As an example of model building on KIEF, we demonstrate how to build an FEM model of the turntable mechanism. For this purpose, we plugged a solid modeler and an FEM solver into KIEF, and prepared a model library for it.
- Modeling workspace
  In the modeling theory stage, general concepts are converted into concepts specific to the modeling theory. We developed a workspace for this process. In case of FEM, by referring to the ontology of FEM such as finite elements and constraints, the system supports a designer to build an FEM model on the workspace. The workspace is used as an interface between a pre-processor of FEM and KIEF.

The implementation is currently at an early stage. We demonstrate how the system will support the designer for building an FEM model of the turntable mechanism.

5.2 Example of Modeling Process

We assume that the following data about the turntable mechanism exists in the system, such as function, behavior, structure and geometric data.

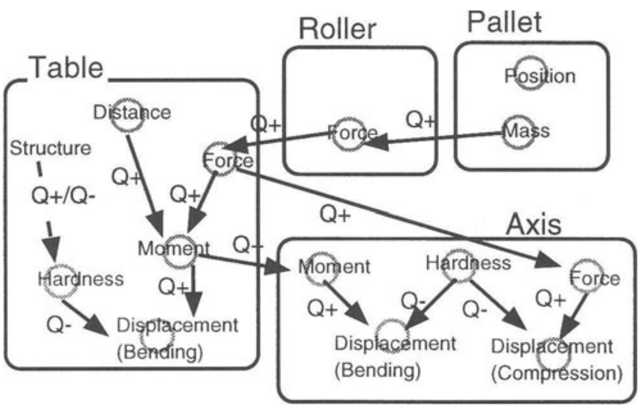


Figure 7: A Parameter Network related to "Displacement"

Figure 6 depicts the function, structure, and behavior of the turntable mechanism.

First of all, the system supports the designer to decide what kind of problem should be analyzed. The design object consists of a turntable mechanism and a roller mechanism (Figure 6a). The system searches each of them from the knowledge base about the goal of analysis, and tells the designer problems which should be analyzed. In this case, the designer pays attention to the problem that is deformation of the table. The system automatically prepares rules related to "Bending," and then reasons out that there is a possibility that table will "bend."

All of the derived phenomena are depicted in Figure 6b. For example, "Force Transmission" means that the load to one object is transmitted to another object.

Next, the designer selects an appropriate analysis method and decides the extent and assumptions for analysis. The definition of a physical phenomenon contains relationships among attributes of entities that the phenomenon occurs to. According to the metamodel with the derived phenomena in the previous stage, the system instantiates attributes related to the phenomena one by one, and derives possible relationships among those attributes by referring to the knowledge about qualitative relationships among the attributes. Figure 7 shows the parameter network generated by this process.

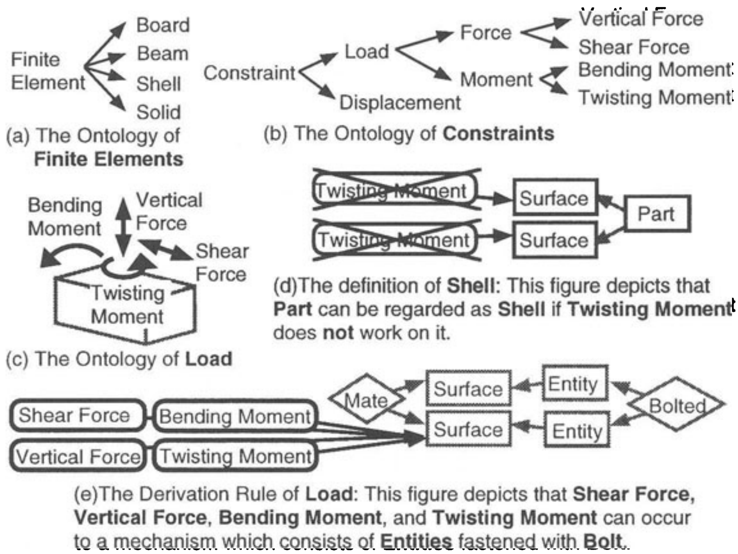


Figure 8: The Ontology of FEM

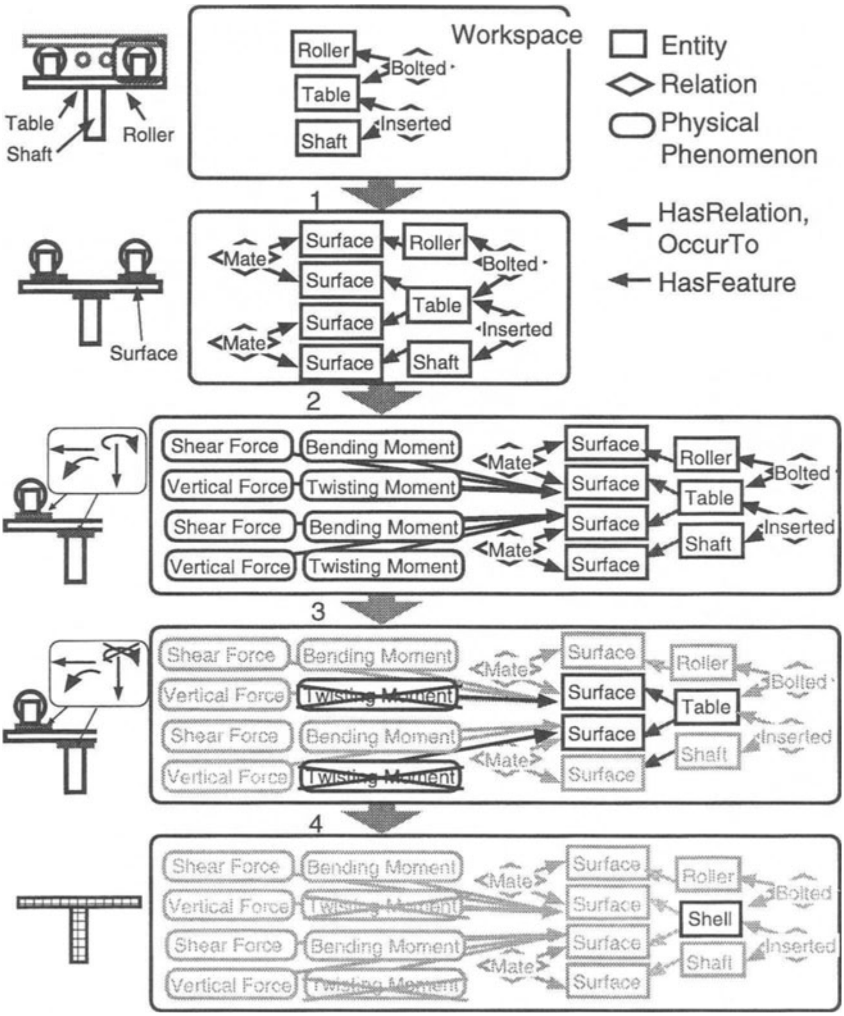


Figure 9: A Process of Finding Appropriate FEM Element for a Table

The designer selects one or a few parameters in the network. In this case, "Displacement of Table" and "Displacement of Axis" are considered. The value of "Displacement of Table" is expected small, because "Pallet" cannot be safely transferred to neighboring cells if "Displacement of Table" is large. Therefore, the designer tells the system to consider increasing of "Displacement of Table" to the model. According to Figure 7, the system proposes a possible way to increase "Force applied to the table" in order to increase "Displacement of Table," and advises the designer to consider the situation that the value of "Force" is large. In this case, the designer decides to analyze the turntable mechanism under the condition that the most load is applied to the table, and to simplify the structure of the turntable not to increase "Hardness of Table."

The designer also has to choose an appropriate tool for structural analysis. The current KIEF system offers FEM and BeamModeler (a simple tool we developed for analyzing strength of a beam based on strength of materials), and the designer can refer to the necessary information to use these specific tools. According to this information, the designer selects a tool.

In the modeling theory stage, the designer converts general concepts in the metamodel to the concepts specific to FEM. The ontology of FEM are defined in the model library for FEM. Figure 8(a) and 8(b) shows the hierarchical relationships among the ontology of FEM. Figure 8(d) is the definition of "Shell."

First, KIEF instantiates concepts about geometry in the workspace (Step 1 of Figure 9), because the ontology of FEM is related to that of geometry. These data can be instantiated by referring to the solid model.

Next, the system derives all forces and moments which will work on each surface by referring to the definition of constraints such as Figure 8(e) (Step 2 of Figure 9). In this case, these forces and moments are derived, because every part is bolted.

After that, the designer specifies some forces and moments that can be neglected. Then, the system proposes an appropriate finite element type for the given conditions with each definition of finite elements such as Figure 8(d) (Step 3 of Figure 9).

In this case, we assume the designer neglects twisting moment that works on the upper surface of the table. Then, the system proposes "Shell elements" for making mesh data of the turntable.

Finally, an aspect model of the FEM model can be built by repeating these steps (Step 4 of Figure 9).

In the tool stage, we build an FEM model on a specific FEM preprocessor (See Figure 10). The system also supports assigning data to the analysis model. For example, since we have to specify two surfaces of a part.

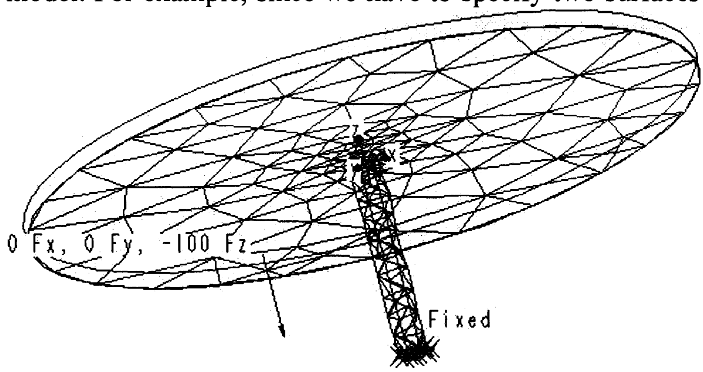


Figure 10: An FEM Model

in order to use "Shell Elements" on the FEM pre-processor we use, the system highlights the surfaces on which the designer neglects twisting moment.

6. DISCUSSION

In this research, we proposed formalized structure of knowledge about modeling processes, which was divided into four stages and also proposed how to support modeling on each stage.

In the design stage, the system detects problems to be considered, and the designer selects problems to be solved. Then, the system automatically extracts the reasoning rules relevant to the problems the designer selected.

Although one of the reasons to support the designer with KIEF is to tell what the designer cannot estimate, it takes considerable time to derive all possible phenomena if there are too many reasoning rules. There are some research reports that are categorized into the area called relevance reasoning in AI (e.g. Levy, 1997). Roughly speaking, it aims at deriving rules relevant only to the current focus for the efficiency of the reasoning. We implemented an algorithm to extract relevant rules according to the idea of relevance reasoning. In case of the turntable, the system works with only rules that have some connection to deformation.

In the analysis stage, to support the designer's decision making, we are interested in using a parameter network which represents qualitative relationships among parameters, and tried to use it to determine modeling assumptions such as neglecting detailed structure of the table and magnitude of load applied to the turntable. Although we believe there should be an algorithm that can make a better use of the network, the way we proposed in this research was simple. A modeling operation often influences a certain parameter both in the positive and negative directions. Therefore, we need a method to evaluate the extent of influences. We have a plan to implement Nayak's algorithm of order of magnitude reasoning (Nayak, 1992).

In the modeling theory stage, we considered that there existed ontology specific to each modeling theory. We studied reusability of universal ontology in our former research (Sekiya, 1996), and we believe the existence of such general ontology for the integration of various tools theoretically. Ontology that is dependent on a modeling theory, but independent of a modeling tool, supports integration of the same sort of tools for a certain modeling theory and representation of modeling knowledge specific to a certain modeling theory.

Choueiry proposes a framework for evaluating reformulation techniques for reasoning about physical systems (Choueiry, 1998). It is worth analyzing KIEF in their framework. Following the terminology they use, we do not have "Strategy" that is a sequence of constructing a specific model. Now we are planning that KIEF provides typical examples referred to when constructing a model. This idea can be categorized into "Strategy."

Ozawa also works on integrating engineering models similar to KIEF (Ozawa, 1998). The main difference between his work and ours is that he is trying to share models on the parameter level, whereas ours is on the conceptual level.

7. SUMMARY

In this paper we described knowledge used in a typical modeling process during engineering design, and we categorized it into four stages. We also described how to support each stage with KIEF. Particularly, in the first and second stages, we proposed methods for starting up the modeling process, and then building a model for analysis.

From a theoretical point of view, we still have to clarify mechanisms to determine a modeling goal. To support the designer to construct a model for the practical use, we have to develop a method to automate routine work of model building and to connect KIEF with other existing analysis tools for cooperative use.

ACKNOWLEDGMENT

This research was funded by the "Modeling of Synthesis" project of the Japan Society of for the Promotion of Science's Research for the Future Program under the contract number, JSPS-RFTF 96P00701.

REFERENCES

Addanki ,S., Cremonini, R., and Penberthy, J. (1991). Graphs of models, *Artificial Intelligence*, Vol. 51, No. 1-3, pp.145-177.

Choueiry, B. Y., McIlraith, S., Iwasaki, Y., Loeser, T., Neller, T., Engelmore, R. S., and Fikes, R. (1998). Thoughts Towards a Practical Theory of Reformulation for Reasoning About Physical Systems, *Working notes of the Symposium on Abstraction, Reformulation, and Approximation (SARA '98)*, Pacific Grove, CA, SA, pp.25-36.

Falkenhainer, B. and Forbus, K. (1991). Compositional modeling: finding the right model for the job, *Artificial Intelligence*, Vol. 51, No. 1-3, pp.95-143.

Fowler, J. (1995). *STEP for Data Management, Exchange and Sharing*, Technology Appraisals Ltd., First edition.

Ishii, M., Sekiya, T. and Tomiyama, T. (1995). A Very Large-scale Knowledge Base for the Knowledge Intensive Engineering Framework, In N.J.I. Mars (eds.): *KB & KS '95, the Second International Conference on Building and Sharing of Very Large-Scale Knowledge Bases*, IOS Press, Ohmsha, Amsterdam, Oxford, Tokyo, and Washington, DC, pp. 123-131.

Levy, A. Y., Iwasaki, Y., and Fikes, R. (1997). Automated Model Selection for Simulation Based on Relevance Reasoning, *Artificial Intelligence*, Vol. 96, No. 2, pp.351-394.

Nayak, P. (1992). Order of Magnitude Reasoning using Logarithms, In *The 3rd International Conference on Principles of Knowledge Representation and Artificial Intelligence*, pp.710-716, AAAI Press, San Jose.

Nayak, P. and Joskowicz, L. (1996). Efficient compositional modeling for generating causal explanations, *Artificial Intelligence*, Vol. 83, No. 1-3, pp.193-227.

Ozawa, M., Cutkosky, M. R., and Howley, B. J. (1998). Model Sharing among Agents in a Concurrent Product Development Team, In Finger, S., Tomiyama, T., and Mäntylä, M. (eds.): *Knowledge Intensive CAD Volume 3, Proceedings of the Third IFIP WG 5.2 Workshop on Knowledge Intensive CAD,* pp. 94-105.

Pos, A. (1997). *Automated Redesign of Engineering Models*, Ph.D. thesis, Twente University, The Netherlands.

Sakao, T., Kondoh, S., Umeda, Y., and Tomiyama, T. (1996). The Development of a Cellular Automatic Warehouse, In *IEEE/RSJ International Conference on Intelligent Robots and Systems '96 (IROS '96)*, pp. 324-331.

Sekiya, T. and Tomiyama, T. (1996). Case Studies of Ontology for the Knowledge Intensive Engineering Framework, In Finger, S., Mäntylä, M., and Tomiyama, T. (eds.): *Knowledge Intensive CAD Volume 2,* Chapman & Hall, pp. 139-156.

Tomiyama, T., Kiriyama, T., and Umeda, Y. (1994). Toward Knowledge Intensive Engineering, In Fuchi, K. and Yokoi, T. (eds.): *Knowledge Building and Knowledge Sharing*, Ohmsha, Tokyo, Osaka, and Kyoto.

Tomiyama, T., Umeda, Y., Ishii, M., and Yoshioka, M. (1996). Knowledge systematization for a knowledge intensive engineering framework, In Mäntylä, M., Tomiyama, T., and Finger, S. (eds.): *Knowledge Intensive CAD Volume 1*, Chapman & Hall, pp. 33-52.

Yoshioka, M., Nakamura, M., Tomiyama, T., and Yoshikawa, H. (1993). A design process model with multiple design object models, In *Design Theory and Methodology (DTM '93)*, ASME, New York, pp. 7-14.