UNIVERSITY OF ZIELONA GÓRA
FACULTY OF MECHANICAL ENGINEERING
in association with the Design Society

THIRD INTERNATIONAL
SEMINAR AND WORKSHOP
10th - 12th October 2002
Zielona Góra - Łagów
POLAND

ENGINEERING DESIGN IN INTEGRATED PRODUCT DEVELOPMENT
Design Methods that Work

ONTOLOGY-BASED MODELING OF PRODUCT FUNCTIONALITY AND USE
PART 1: FUNCTIONAL-KNOWLEDGE MODELING

Y. Kitamura, R. Mizoguchi

Osaka University
The Institute of Scientific and Industrial Research
e-mail: {kita,miz}@ei.sanken.osaka-u.ac.jp

Keywords: Process modeling, knowledge modeling, functionality, ontology

Abstract: *Although importance of knowledge sharing among designers has been widely recognized, the knowledge about functionality in the conceptual design phase is often scattered across technical domains and it lacks consistency. Aiming at capturing such functional knowledge consistently, we have developed a framework for its systematic description based on the functional ontologies, which provide fundamental concepts for capturing the target world and a common vocabulary for description of functional knowledge applicable to other domains. A successful deployment of our framework in a production company is discussed. We also mention a design supporting system using the systematized knowledge. The second part of this twofold paper presents a collaborative research with Delft University of Technology elaborating on use and unintended behavior. This first part introduces the basis of extension and discusses further issues which are found in the collaborative research.*

1. INTRODUCTION

In order to speed up the design process and to improve the quality of products, knowledge sharing among designers is important. Although the advancement of computer technologies has enabled easy access to information related to the structure and/or the shape of artifacts using CAD on computer networks, it is difficult to share the design know-how, including knowledge about functionality, during the conceptual design phase. Such knowledge is often scattered across technical domains and it is improperly categorized as we will discuss in Section 2. Systematization of such knowledge according to certain principles enables designers to access knowledge from the different domains.

The main goal of this research is to promote sharing of the functional design knowledge among human designers by providing a framework for systematic description of the functional knowledge based on Ontological Engineering [1]. Ontologies can provide fundamental concepts for capturing the target world in a consistent way and they can provide a vocabulary for description of knowledge. Such concepts help us improve consistency and generality of knowledge. We have developed two ontologies for functional knowledge, namely, an extended device ontology and a functional concept ontology [1,2,3].

This paper discusses ontology-based modeling of the functionality of products and a systematic description of functional knowledge. Firstly, we analyze difficulties in description of the functional knowledge and discuss the need for ontologies in Section 2. After an overview of our functional modeling framework in Section 3, Section 4 shows the functional concept ontology which contains concepts representing functionality of components. Next, in Section 5, we propose a new key concept for consistently capturing functional knowledge about the achievement of functions, called "way of function achievement". Section 6 discusses utilities of the ontologies in describing functional knowledge.

Section 7 presents a successful deployment of our framework in a production company and we analyze the success factors. We also discuss a design supporting system using such knowledge in Section 8. It helps human designers (re)design products by providing a wide range of ways of achievement.

The second part of this twofold paper presents a collaborative research with Delft University of Technology elaborating on use and unintended behavior. This first part introduces the basis of extension and discusses further issues which are found in the collaborative research in Section 9.

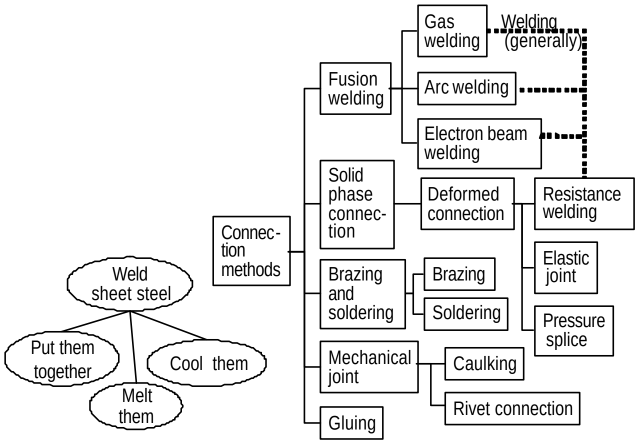
Section 10 discusses related work as well as limitations and application domains of our ontologies followed by concluding remarks.

2. THE NEED FOR ONTOLOGIES

In this paper, the target knowledge is conceptual design knowledge concerning functionality. Functionality plays a crucial role in the conceptual design of engineering products [4,5,6,7]. A designer often decomposes a required function into sub(micro)-functions by means of a so-called functional decomposition [4] using knowledge about how to achieve the required function. We call such knowledge, that represents achievement relations of functions, functional knowledge. Because many inventions are based on a new combination of known techniques and/or use of known techniques in different domains [8], the innovative design capability of designers is augmented, if they are supported by a knowledge-base containing a wide range of such functional knowledge from different domains.

However, it is not easy to describe such functional knowledge about functional decomposition in a way that is consistent and applicable to other domains. Figure 1 shows examples of inadequate descriptions. Firstly, as one can represent a function as a transitive verb (verb plus target objects) as it is done in value analysis [9], one might describe a functional decomposition as is shown in Figure 1(a) as a functional model of a manufacturing facility to make two sheet steels connected. It suggests that the function "to weld sheet steel" can be achieved by the sequence of the three sub-functions. However, "to weld sheet steel" is a composite of the two phenomena; the sheets are connected and a part of the sheets is fused. From the functional point of view, the former is the intended goal, while the latter is just a feature of the specific way to achieve the goal. In fact, the same goal can be achieved in different ways (e.g., using bolts and nuts) without fusion. To allow freedom in design and to make selection of "bolt & nut" instead of "welding" possible, the achieved function should be the same; "to connect".

This example (Fig. 1(a)) shows us importance of conceptualization of both "what is achieved" and "how to achieve". The former is often called "goal" or "function" and has been investigated in engineering design [4,10], value engineering (VE) [9, 11] and functional reasoning [e.g., 6,7,12,13,14]. As pointed out in [11], a common vocabulary of function is necessary for reuse of functional knowledge in different domains. Moreover, it should be machine understandable in order for a computer to manage the knowledge. Nevertheless, only a few generic functions have been proposed to date [4,13,14]. They have a very high level of abstraction. For example, very few (from 3 to 24) generic functions are defined in [4]. Although Tejima et al. have developed a standard set of 158 verbs representing function for VE [11], the definitions of the verbs are only for human comprehension. We need a rich and comprehensive vocabulary of functional concepts with operational definitions



(a) A functional decomposition

(b) A categorization of connection methods [16] (Translated by the authors)

Figure 1. *Inadequate descriptions of functional knowledge*

Likewise, little is known concerning principles or guidelines for how to describe the latter, "how to achieve a function". So-called design catalogs mainly concentrate on concrete mechanical pairs. Although the general knowledge about the issue is described in design literature (e.g., [6,15]), there is neither a rule nor a principle of how to formulate such knowledge and thus the description tends to be ad hoc. As a result, we found many inadequate categorizations of "how to achieve". For example, the categorization of connection-methods shown in Figure 1(b) that was found in a textbook published by an academic society [16] is not consistent and ill-structured. The first-level categorization is not consistent according to different characteristics (the phase for the upper two and the binder for the lower three). The upper-right second-level categorization is ill-structured as is shown by the irregular dashed links, because the three categories do not represent methods of connection but methods of heating.

The needs for consistent and sharable description of functional knowledge can be summarized as "fundamental and generic concepts for capturing and describing the functional knowledge". Such specification of conceptualization is generally called an "ontology" [1,17]. In its role as meta-knowledge [1], an ontology can provide constraints and/or conceptual principles with knowledge authors. Thus, our idea to reduce the difficulties discussed thus far is to define the following concepts as ontologies;

- A common vocabulary of concepts representing functions of engineering devices.
- Fundamental concepts for capturing the target knowledge, especially, how to achieve functions.

These are discussed in Section 4 and 5, respectively after an overview of our modeling framework.

3. A MODELING FRAMEWORK OF FUNCTIONALITY

This section overviews our approach for functional models of artifacts and framework of them as summarized as shown in Figure 2. This framework is an extension of our functional modeling language FBRL (abbreviation of a Function and Behavior Representation Language) [18]. There are three axes denoted a, b and c. The vertical axis denoted (a) represents the grain size of the entities among which there are whole-part (aggregation) relations. The second axis to the depth denoted (b) represents relations among entities of the same grain size. Lastly, the horizontal one denoted (c) represents the objective-teleological interpretation, that is, from behavioral level (including structure) to functional level. Between these levels of an instance model, there are the ontologies and generic knowledge. The extended device ontology provides fundamental concepts for both levels. The functional concept ontology makes mappings between behaviors and functions. The knowledge-base containing generic ways of function achievement provides building blocks for the functional level.

3.1. Behavioral Level

The model at the behavioral level is objective without intention of designers. We adopt the device-centered view of artifacts which regards any artifact as composition of devices (components) which process input to produce output as a black box. A device is connected to another device through its input or output ports (as shown in the depth axis (b) in Figure 2). A device consists of other devices of smaller grain size and usually is organized in a whole-part hierarchy of sub-devices (as shown in the vertical axis (a)). We discuss this device-centered view in detail in Section 6.2.

A device changes states of things inputted which are called *operands* such as substance like fluid, energy like heat, motion, force and information. The difference between the states of the operand at the input port and that at the output port is called as "behavior" of the component.

A behavioral-level model of a target system in FBRL [18] is a set of model of components (devices) representing behavior (mode) of each component, connection information among ports of the components and structural hierarchy of the components. A model of a behavior consists of objects as operand, qualitative constraints over attributes of the objects (called QN-relations), and material-product relations each of which represents an output object is made of specific inlet object(s) (called MP-relations). Consider a heat exchanger between two streams of water shown in Figure 3 as an example. It has four ports connecting to neighboring components and operands including inlet/outlet waters and thermal energies on them. Its behavioral model is described as qualitative constraints (QN-relations) such as those over the temperatures of the water flows as a medium of thermal energy and the heat resistance, and material-product relations (MP-relations) such as the outlet thermal energy at the port No. 2 (denoted p2) is made of inlet one at the port No.1 (p1) and a part of inlet one of the port No.3 (p3).

3.2. Functional Level

A (*base-)function* of a component at functional level is defined as a result of teleological interpretation of a behavior of the component under an intended goal [18]. We consider verbs representing functions of components to be (base-) functional concepts. As shown in Figure 2, these functions in a functional model of a specific target system are instances of generic functions in the functional concepts ontology. A function in the model is associated with a behavior realizing it by the mapping primitives called as the *Functional Toppings* (FTs) of FBRL [18]. The functional toppings represent information about teleological interpretation of behavior according to the designers' intention.

For example, when the heat exchanger shown in

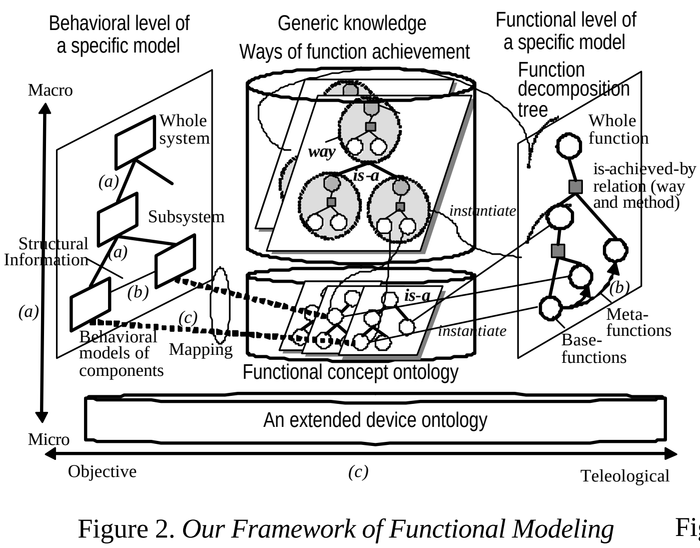


Figure 2. *Our Framework of Functional Modeling*

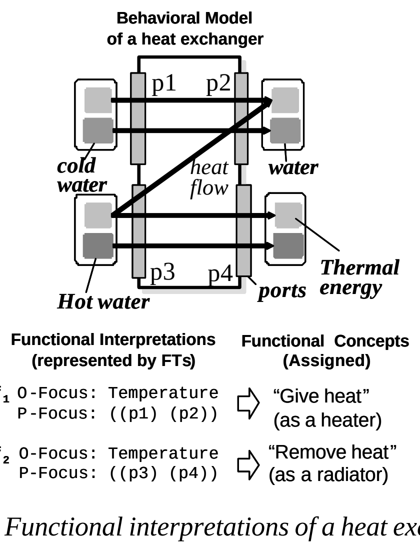


Figure 3. *Functional interpretations of a heat exchanger*

Figure 3 is used as a heater or a radiator, it performs different functions ("heat the cold water" or "cool the hot water", respectively). The behavior is, however, the same. The functional toppings in Figure 3 show the difference between these two functional interpretations and they are automatically matched with functional concepts in the functional concept ontology. The details of definitions of functional concepts and functional toppings are discussed in the next section.

Another point in Figure 2 is generic knowledge of how to achieve a function, called "ways of function achievement". The "is-achieved-by" relations in the function decomposition tree specific to a target product (the right part in the figure) are instances of generic ways of function achievement (the upper-center part). The details of the knowledge are discussed in Section 5.

Each base-function has a specific type of functions (called *function types*) and plays a role for another function (called *meta-functions*). The function types represent causal patterns of achievement for goals of each base-function of a component such as ToMake and ToMaintain (we redefined the ones in [5]). The meta-functions are result of teleological interpretation of causal relations among base-functions of different components (i.e., inter-components relations) such as ToEnable and To-Provide. They represent collaborative roles in inter-dependency among base-functions. For more detail on meta-functions, see [2].

4. A FUNCTIONAL CONCEPT ONTOLOGY

Figure 4 shows a portion of the functional ontologies implemented in our environment for building and using ontologies named Hozo [19]. The left window in Figure 4 shows a portion of the functional concept ontology [2]. It consists of about 120 base-functions for functions of devices, 4 function types and 8 meta-functions mentioned above. The base functions are categorized by kind of target operand (things to be changed by the function) such as substance, energy, information, force and motion. The left window in Figure 4 shows a portion of an *is-a* hierarchy of the energy-related base-functions.

Each concept in the hierarchies is operationally defined using the FTs (Functional Toppings) of FBRL. The FTs represent information about the teleological interpretation of (mapping to) a behavior according to the designers' intention. A set of FTs can be composed of the following items;

??*Obj-Focus* specifies kind of operands (such as substance and energy) which the designer intended to change (operand to focus on).

??*O-Focus* specifies types of physical attributes (such as temperature and amount) of the operand to focus on.

??*P-Focus* specifies ports (interaction with neighbor components through the ports) to focus on

??*Necessity* specifies the necessity of the focused operand.

For example, an energy function, *To shift energy*, is defined as a behavioral constraint: *focused energy moves between two different mediums*. It can be defined by the axioms inherited from the super-concept plus the following three axioms;

??P-Focus on an inlet port and an outlet port

??Energy in the focused outlet port is made from energy in the focused inlet port.

??Mediums of the focused energies are different.

The right window in Figure 4 shows the window for describing the second axiom. The "participant" field defines arguments (parameters) appearing in the axiom body, "axiom body" defines constraints over arguments. The constraint should be true for all instances of the class if the "condition" is true (the symbol "T" means "always true").

To take, a subtype of *to shift* in the *is-a* hierarchy, is defined as FTs of *to shift* with an additional FT, P-Focus on the port of energy provider. Likewise, *to remove* is defined as that of *to take* with an additional FT, the energy taken is *unnecessary* as *Necessity* FT.

5. KNOWLEDGE ABOUT WAYS OF FUNCTION ACHIEVEMENT

5.1. The Concept of "Way of Function Achievement"

When a function is achieved by a sequence of sub(micro)-functions, we call such a relation a

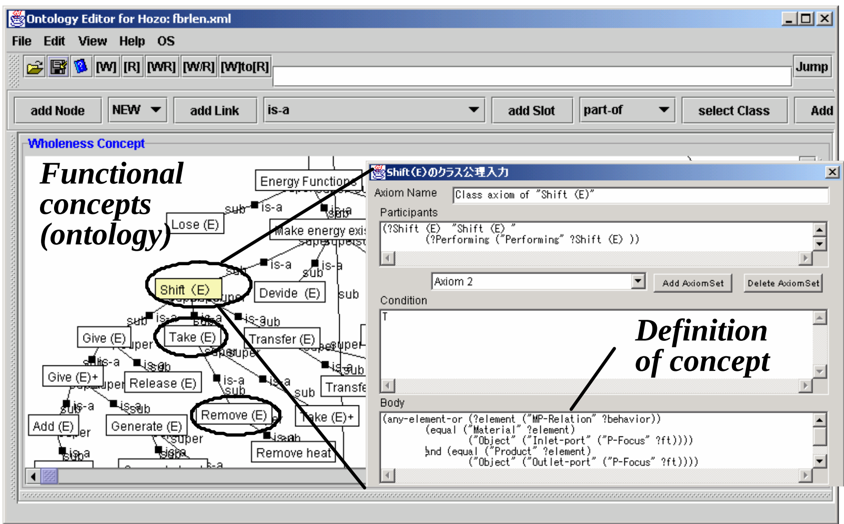


Figure 4. *A functional concept ontology (portion)*

"function achievement relation". A tree (hierarchy) of function in such relations is called *a function decomposition tree*. Figure 5 shows a part of the function decomposition tree of a coffee maker as an example. The required function "to make coffee" is decomposed into "to extract coffee" and "to keep coffee". The sub-functions are further decomposed into finer-grained micro-functions.

Such a traditional functional decomposition model represents only "how" the macro-function is achieved but does not represent "why" the sequence of the sub-functions can achieve the macro-function.

Here we introduce the concepts of *method of function achievement and way of function achievement*. We call the sequence of sub-functions *the method of the achievement*. On the other hand, background knowledge of functional decomposition such as physical principles, theories, phenomena, and structure as the basis of the achievement is called the *way of the achievement*. For example, in Figure 5, the basis of the second functional decomposition can be represented as "indirect-heat fluid way" for which the ingredient (coffee taste) is extracted by contacting the target object (ground coffee) with hot fluid (water).

We call general knowledge of function achievement *the way of function achievement*. Its description consists of a macro-function, a set of sub(micro)-functions, temporal and causal constraints among sub-functions, principles of achievement, conditions for use of the way, and characteristics of operands using the way. Although it includes a description of the method of function achievement, we call it *a way of function achievement*, focusing on the fact that it includes description of principle of the achievement.

5.2. Utility of the "Way" Concept

The concept of way of achievement helps us to detach "how to achieve" (way) from "what is achieved" (function). For example, as mentioned in Section 2, "to weld" is not just a function but function with a way in which the target is melted. It should be decomposed into the "connecting function" and "fusion way". This increases generality and capability to cover wide range of ways such as the bolt and nut way as an alternative to connect.

As another example, the "arc welding" way shown in the ill-structured part of Figure 1(b) should be decomposed into two different ways, that is, the fusion way for connection and the arc way for heating.

5.3. The *is-a* Hierarchy of Ways

The ways of achievement of a function are organized as an *is-a* hierarchy shown in Figure 2 according to the physical principles on which they are based. Because the principles are inherent properties

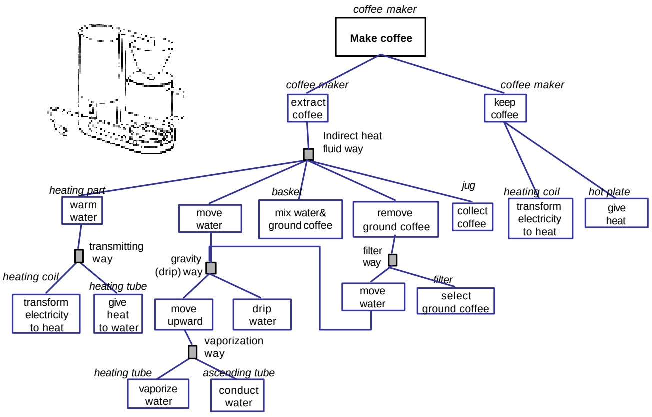


Figure 5. *A function decomposition tree of a coffee maker (portion)*

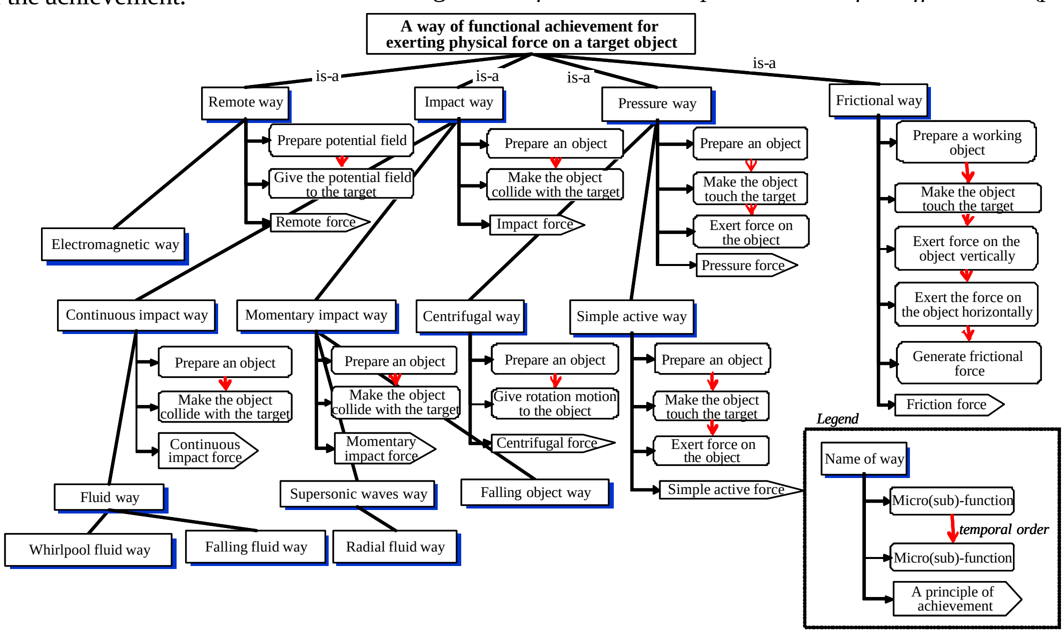


Figure 6. *An **is-a** hierarchy of ways for exerting force (portion)*

of the ways, we can consider them to be organized in a straightforward *is-a* hierarchy. As an example, Figure 6 shows an *is-a* hierarchy of ways of achievement for "exerting physical force". In the figure, a box, a round box, and a pentagon represent a concept of way, a sub(micro)-function of the macro-function in the way, and a principle of the way, respectively.

Note that the three types of trees of functions in Figure 2 are different from each other. The function decomposition tree represents is-achieved-by (a kind of *part-of*) relations among functions. The *is-a* hierarchies of ways represent an abstraction of the key information about how to achieve the function, while the *is-a* hierarchies in the functional concept ontology discussed in Section 4 represent abstractions of functions themselves, that is, the goals that are achieved. Moreover, the numbers of the ways for a function are unlimited in nature, while the numbers of functional concepts are small.

6. UTILITIES OF ONTOLOGIES

The ways of function achievement and functional models of the target product are supported by two functional ontologies, that is, the functional concept ontology and the extended device ontology. In this section, we discuss their utilities.

6.1. Utility of the Functional Concept Ontology

The functional concept ontology discussed in Section 4 provides vocabulary for functional models and knowledge. As shown in Figure 2, functions in a functional model of a specific target system are instances of the generic functions.

The knowledge about ways of function achievement is also described in terms of functional concepts defined in the ontology. The definitions using FTs scarcely depend on the device, the domain or the way of its implementation, so that they are very general and usable in a wide range of areas.

All entities in the functional level are defined operationally using functional toppings in the functional concept ontology. By "operational", we mean that these functional concepts can be mapped automatically from behaviors. In fact, we have developed an automatic identification system which generates plausible functional interpretations of given behaviors [2]. In other words, the functional concept ontology specifies the space of functions and limits functions within the generic functions defined in the ontology. The automatic identification system screens out the candidates of functional interpretations which match with no functional concept in the ontology on the assumption of comprehensiveness of our ontology. This assumption is discussed in Section 10. Although it may reduce the freedom of functional representation in comparison with hand-written functional models, automatic identification of functional concepts enables us to avoid ad hoc modeling and obtain consistent functional models.

6.2. Utility of the Device Ontology

The aim of introducing the device ontology is to impose a proper viewpoint from which one can successfully model a system in various domains in a way that the viewpoints are consistent with each other. It is not an easy task to build models of a lot of artifacts in a consistent way. For example, the concepts "a gear pair changes torque", "a cam shrinks a spring" and "a spring pushes up a rod" in a product model are inconsistent with each other in the hidden computational models. While the first functional statement "to change torque" assumes the torque flowing into the component (the gear pair), the latter two statements represent changes of entities (the spring and the rod) which do not flow into the components (the cam and the spring, respectively). Moreover, the roles of the spring in the second statement and the third statement are different (the operand (target) role and the agent (actor) role, respectively). If one uses these statements in a model without being aware of such differences, the consistency of the model will be lost in a strict sense. Using the same framework to capture the target world is necessary for consistent and interoperable models. The (extended) device ontology mentioned below aims at building interoperable models without inconsistency and providing a guideline for modeling process.

The device ontology specifies a device-centered view of artifacts that regards any artifact as a composition of *devices* which are considered as something that plays the main actor (*agent*) role in changing the input into the output. The things changed as the input and the output are called *operands* here.

From our point of view, the device ontology specifies the *roles* played by the elements that collectively constitute a device. The concept of role is a hot topic in ontological engineering because an entity plays different roles in different situations, and disregarding this fact has been a major source of failure in conceptualization of the world [1].

A naïve idea of the device ontology was born in system engineering. It is composed of components and connections between them and it has been extensively used in many engineering areas as well as in design community [4]. However, it has no criterion of which role should be played by which component and the assumptions behind the ontology are implicit, and therefore modeling of artifacts can be ad-hoc. Even worse, it is hard to compare it with other ontologies and its limitations are not clear.

De Kleer and Brown introduced an idea of *conduit* into the naïve device ontology [20]. A conduit can be defined as a special type of device that can be considered as it transmits an operand to output port without any change in an ideal situation. However, their ontology still leaves the identification of operands that is affected by *agents* unclear. Umeda et al. points out the limitation of functional decomposition based on device ontology [6] and Mortenesen reports on a negative observation on the applicability of device ontology to mechanical elements [21].

Aiming at clear specification of roles of entities for consistent modeling and covering mechanical domains as well, the authors proposed an extended device ontology which includes four different concepts of behavior and introduces the concept of *medium* [1, 3]. A medium is something that holds an operand and enables it to flow among devices. For example, steam can play the role of a medium because it can hold heat energy. We allow such a role sharing that a conduit plays the role of a medium at the same time. For example, while a shaft is a conduit for force and motion, at the same time, it plays the role of medium for them. For more details on the extended device ontology, see [3].

Note that our claim is **neither** that the device ontology enables all kinds of descriptions for all kinds of artifacts, **nor** that all models should be described on the basis of the device ontology, **nor** that the device ontology is an unique solution. We claim that a solid foundation like this device ontology is needed for systematic description of functional knowledge.

7. DEPLOYMENT IN INDUSTRY

7.1. Deployment in Production Systems

Our framework is being deployed in the Production Systems Engineering Division of Sumitomo Electric Industries for sharing functional design knowledge of production systems. After one year study of our theory, the company started test use in May, 2001. Sumitomo engineers and we described function decomposition trees for about 15 production facilities in production systems for semiconductors. As an example, Figure 7 shows the function decomposition tree of a wire saw for cutting ingots.

7.2. Building a Knowledge-base of Ways

We have described 104 generic ways of achievement for 26 functions from five examples; a washing machine, a printing device, slicing machines for ingots of semiconductors (using a wire or a rotating blade), and an etching device. Firstly, we described function decomposition trees of the example artifacts. Next, we generalized the ways appearing in the examples. Then, we tried to find the underlying principles and then organized them into *is-a* hierarchies as discussed above. Lastly, we added other alternative ways from other technical domains based on the principles extracted. The *is-a* hierarchy shown in Figure 6 has been built in such manner, and is commonly used in the examples. For example, the fluid way for exerting force is used not only in washing machines but also in the slicing machine for removing scrapings.

7.3. Effects and Evaluation

In general, it is difficult to evaluate such a framework for knowledge systematization strictly, because there is no quantitative measure for quality and generality of knowledge-base. Here, we report empirical evaluation of the deployment in Sumitomo Electronic Industries. We also discuss the limitation of our ontologies in Section 10.

The preliminary evaluation by the Sumitomo engineers was unanimously positive. They said that this framework enabled them to explicate the implicit knowledge possessed by each designer and to share it among team members. It was easy for designers to become familiar with the framework based on the device ontology. They decided to deploy it and started the development of a knowledge collecting software.

The followings are some of remarkable suc-

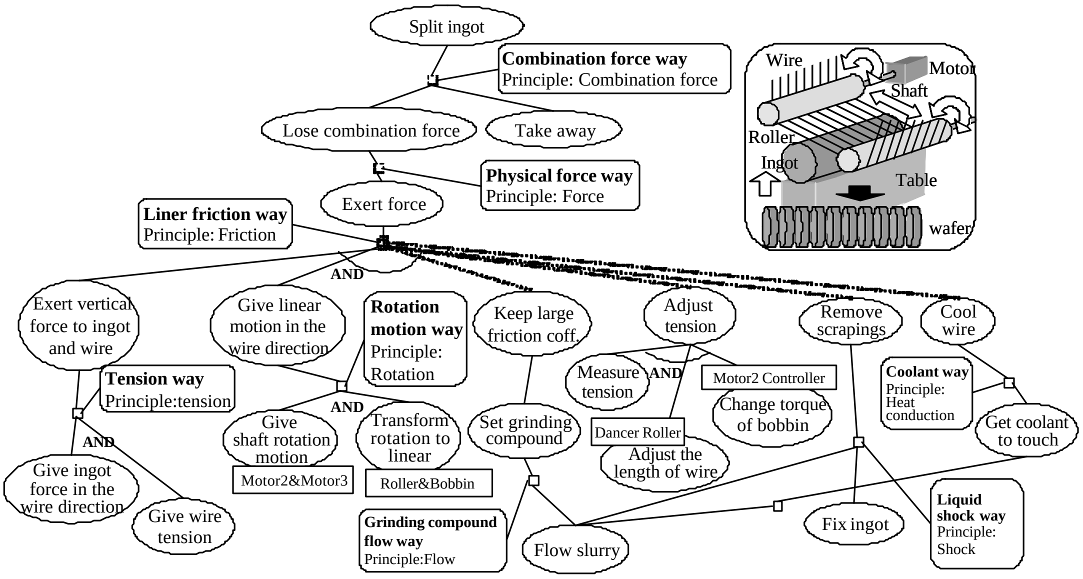


Figure 7. *A function decomposition tree of a wire-saw (portion)*

cesses in the test use;

- ✍✍A designer was not able to solve a problem of low quality of semiconductor wafers after 4-month investigation. By exploring causes of the problem in the model of ways of function achievement with a clear description of physical principles, he found a solution for the problem within 3 weeks.
- ✍✍The models of ways of function achievement were used as *knowledge media* for collaborative work by people having different viewpoints such as manufacturing engineers, manufacturing equipment engineers, equipment operators and equipment maintainers. Although mutual understanding and collaboration among them was strongly required, it never happened. The use of our framework, however, enabled them understand and collaborate with each other in a facility improvement project. It turned out that the framework worked as a common vocabulary which lacked before.
- ✍✍A feasible new improvement of the wire-saw was found from the knowledge-base by adopting the way of using magnetic fluid for controlling tension of the wire. This can be done by applying a way originating from the textile industry to the semiconductor industry. This indicates feasibility of our framework for general functional knowledge.

The success factors of the deployment can be summarized as follows; (1)clear discrimination between function (goal) and way (how to achieve the goal) which contributes to reusability of the knowledge, and (2)clear discrimination among *is-a* and *part-of* relations, that is, the *is-a* hierarchy of functions and that of ways, and the *is-achieved-by* (a kind of *part-of*) hierarchy of function, and (3)explicit viewpoint specification by the extended device ontology.

8. DESIGN SUPPORT USING FUNCTIONAL KNOWLEDGE

This section presents a design support system named "functional way server" as an application of the functional knowledge that is described based on ontologies. The server contains the knowledge of ways of function achievement as *is-a* hierarchies. The server is designed to support conceptual (re)design of engineering devices by providing various kinds of ways of function achievement which can be a hint for possible alternatives. In conceptual redesign of existing artifacts, designers can change a specific way of function achievement in the original design based on alternative ways provided by the server.

For example, in a certain type of washing machine, the main function "to remove dirt from cloth" is achieved by friction caused by rotating a screw in the water. Given the associated function decomposition tree and "reduction of damage of cloth" as a new requirement, a designer can get a new function decomposition tree by selecting the centrifugal way from the ways for exerting force (its *is-a* hierarchy is shown in Figure 6). This type of washing machine was recently introduced in the markets with the aim to reduce damage of cloth. The server provides a wide range of ways from different domains and then facilitates innovative design.

9. FURTHER RESEARCH

This section discusses remaining research issues of which importance is recognized in the collaborative research. A part of the issues concerned with user actions is discussed in the second-part paper.

9.1. Change of Roles

As pointed out in Section 6, role assignment to entities should be made explicit for consistent modeling. For example, consider a manufacturing process of a product. The components of the product play a role of agent in the functional model when the product works. On the other hand, needless to say, the states of the components are changed as operands in its manufacturing process.

The similar situation is occurred when we consider user's actions when the product is used. Generally speaking, a user's action for a product is to change states of components of the product as the same as activities in the manufacturing processes. The models with user actions are discussed in the second-part paper.

The role assignment in different processes can be considered as a kind of relationship among processes. One of other important relationships is the *invoking relation* in which an action in a process starts another process. Categorization of such relationships is under investigation.

9.2. Functions for Undesirable States

Many products include functions for *undesirable states*. For example, the wire-saw machine shown in Figure 7 includes the function "to remove scrapings". This function is needed for avoiding stuffing of scrapings which are made by cutting ingots. Here, the existence of scrapings is *undesirable state* for the intended functions.

Such functions for undesirable states can be considered as a type of *supplementary functions*, which are not needed for achieving the main function in principle. Other types of them include functions for efficiency of the process (e.g., time and cost of the processes) and quality of the operands. In order to represent design rationale of such functions we are investigating on a modeling framework of undesirable state and the supplementary functions.

The undesirable states can have many causes. In the case of scrapings, the cause is normal execution of the intended function. One of the other causes is a user's action which is not intended by the designers. A designer can add a supplementary function for avoiding the unintended action itself and/or the harmful effects caused by the action. In the second-part paper, such functionality will be discussed.

10. RELATED WORK AND DISCUSSION

Ontologies of artifacts

The importance of explicit specification of conceptualization (i.e., ontology) in artifact modeling is widely recognized in literature such as [7, 22, 23, 24]. As pointed out in these papers, ontological statements (commitment) can be used as simplifying assumptions to improve the robustness, complexity, and computability of the knowledge representation of artifact models, and to avoid misinterpretation of the models.

Chandrasekaran and Josephson try to clarify several interpretations of the concept "function" and discuss relationship between environment-centric functions and device-centric functions based on ontological considerations [7]. Although we share their distinction between the two major views of functions and the attitude towards the ontological analysis, we concentrate only on the device-centric viewpoint in this paper. We define meaning of fundamental concepts such as "device" and "operand (object)" as the roles played by the entities behind the device-centric viewpoint [1, 3].

Borst et al. propose a hierarchy of ontology for designing an artifact which shares a lot of with our work [22]. However, Borst's ontologies do not include one for function which is our main issue.

Horváth et al. discuss design concept ontologies as a part of a comprehensive methodology for handling design concepts in conceptual design, which include structure and shape as well as functionality [23]. Here we concentrate on functionality.

Knowledge of function achievement

In the literature on design, general knowledge for functional decomposition or functional synthesis is proposed (e.g., [6, 15]). The major advantages of our knowledge representation of the ways of achievement include explicit conceptualization of "way", organization in *is-a* hierarchies based on principles of ways, and use of the functional concept ontology as follows.

Firstly, our ways of function achievement are explicit conceptualization of the feature of achievement such as theory and phenomena at the behavioral level. Such functional knowledge is compliant with the observations found in the research on design processes [25] in which it is claimed that functional decomposition is not done solely in the functional space but also by going back and forth between the functional, behavioral and structural spaces. During this process, portions of the artifact are determined in each of the spaces simultaneously.

Secondly, we organized such general knowledge as an *is-a* hierarchy. Although the feature knowledge is also captured by Malmqvist [10], but he focuses strictly on the function decomposition tree of a specific product and there is no organization of general knowledge. In our framework, the principle of function achievement can be conceptualized as a way of function achievement. Such conceptualization helps us to organize the knowledge in consistent *is-a* hierarchies.

Lastly, our functional knowledge is based on a functional concept ontology [2]. Use of such functional concepts as a vocabulary for the description of knowledge facilitates reuse of the knowledge in different domains.

Such knowledge can facilitate innovative design, because many innovative designs are based on techniques known in different domains [8]. TechOptimizer [26] is a software product based on a theory for innovative design [8], which contains generic principles of invention. However, it just searches highly abstract principles based on given criteria.

Limitation of our ontologies and application domain

We cannot claim completeness of the concepts in our functional concept ontology. Note that we did not define domain-specific functions but general functions that are common in many domains. Although one might think that the set of functional concepts is huge, not the set of function but of the set of ways of function achievement is very large. In fact, in Value Engineering research [11], 158 verbs are proposed as a standard general set for representing functions of artifact. Although it includes functions for human sense as well, we concentrate on functions changing physical attributes. Our functional concepts about information are under more precise investigation.

The ontologies have been applied to modeling of a power plant, an oil refinery plant, a chemical plant, a washing machine, a printing device, and manufacturing processes. Their models include changes of thermal energy, flow rate, and ingredients of fluid, force and motion of objects.

The current functional concept ontology can describe simple mechanical products, though it does not cover static force balancing and complex mechanical phenomena based on the shape of objects.

11. CONCLUSION

The contribution of this research can be summarized as framework for description of sharable design knowledge about functional decomposition. In this paper, we discussed the concept of "way of function achievement" and the utilities of functional ontologies as success factors in the deployment in the production company.

We showed feasibility of reusable functional knowledge based on the functional concept ontology. However it is not easy to generalize ways in the concrete functional models. An investigation on guidelines for description of ways is in progress.

Acknowledgements: A part of this work, the model of a coffee maker and the research issues discussed in Section 9, has been explored as a part of the collaborative research with Mr. Wilfred van der Vegte and Prof. Imre Horváth, Delft University of Technology. The authors also wish to thank the former for his valuable comments on the whole of this paper. The authors would like to thank Toshinobu Kasai, Kouji Kozaki, Mariko Yoshikawa and Tomonobu Takahashi for their contributions. Special thanks go to Dr. Masayoshi Fuse, Sumitomo Electric Industries Ltd., for his cooperation for the deployment. This research was supported in part by the Japan Society for the Promotion of Science (JSPS-RFTF97P00701).

References

[1] Mizoguchi, R., and Kitamura, Y., Foundation of Knowledge Systematization: Role of Ontological Engineering, *Industrial Knowledge Management - A Micro Level Approach*, Rajkumar Roy Ed., pp.17-36, Springer-Verlag, London, 2000

[2] Kitamura, Y., Sano, T., Namba, K., and Mizoguchi, R., A Functional Concept Ontology and Its Application to Automatic Identification of Functional Structures, *Advanced Engineering Informatics (former Artificial Intelligence in Engineering)*, Vol. 16, pp. 145-163, 2002.

[3] Kitamura, Y. and Mizoguchi, R., An Ontology-Based Framework for Systematization of Functional Knowledge and Its Use for Supporting Designers, *AI-TR-2002-1*, The Institute of Scientific and Industrial Research, Osaka University, 2002. Also available at http://www.ei.sanken.osaka-u.ac.jp/~kita/pub/TR-2002-1.pdf

[4] Pahl, G., and Beitz, W., *Engineering Design - a Systematic Approach*, The Design Council, 1988.

[5] Keuneke,A.M., Device Representation: the Significance of Functional Knowledge, *IEEE Expert*, 24:22-25, 1991

[6] Umeda, Y., Ishii, M., Yoshioka, M., Shimomura, Y., and Tomiyama, T., Supporting conceptual design based on the function-behavior-state modeler, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 10, 275-288, 1996.

[7] Chandrasekaran, B., and Josephson, J. R., Function in Device Representation, *Engineering with Computers*, 16(3/4), 162-177, 2000.

[8] Sushkov, V.V. Mars, N.J.I., and Wognum, P.M., Introduction to TIPS: a Theory for Creative Design, *Artificial Intelligence in Engineering*, 9, 1995.

[9] Miles, L. D., *Techniques of value analysis and engineering*, McGraw-hill, 1961.

[10] Malmqvist, J., Improved Function-means Trees by Inclusion of Design History Information, *Journal of Engineering Design*, 8(2):107-117, 1997.

[11] Tejima, N. et al. eds., Selection of Functional Terms and Categorization, *Report 49*, Soc. of Japanese Value Engineering (In Japanese), 1981.

[12] de Kleer J., How Circuits Work., *Artificial Intelligence*, 24, 205-280, 1984.

[13] Chittaro, L., Guida, G., Tasso, C., and Toppano, E., Functional and Teleological Knowledge in the Multimodeling Approach for Reasoning about Physical Systems: A case Study in Diagnosis, *IEEE Transactions on Systems, Man, and Cybernetics*, 23(6), 1718-1751, 1993.

[14] Lind, M., Modeling Goals and Functions of Complex Industrial Plant, *Applied Artificial Intelligence*, 8:259-283, 1994.

[15] Bradshaw, J. A., and Young, R. M., Evaluating Design using Knowledge of Purpose and Knowledge of Structure, *IEEE Expert*, 6(2):33-40, 1991.

[16] J. Soc. for Tech. of Plasticity, *Joining - Detailed techniques and possibility* (In Japanese), Corona Pub., Tokyo, 1999.

[17] Gruber, T. R., A Translation Approach to Portable Ontology Specifications, *Knowledge Acquisition*, 5(2):199-220, 1993.

[18] Sasajima, M.; Kitamura, Y.; Ikeda, M.; and Mizoguchi, R. FBRL: A Function and Behavior Representation Language. *In Proc. of IJCAI-95*, 1830-1836, 1995.

[19] Kozaki, K., Kitamura, Y., Ikeda, M., and Mizoguchi, R., Development of an Environment for Building Ontologies which is based on a Fundamental Consideration of "Relationship" and "Role", In *Proc. of The Sixth Pacific Knowledge Acquisition Workshop (PKAW 2000)*, pp.205-221, 2000

[20] de Kleer, J. and Brown, J. S., A Qualitative Physics Based on Confluences, *Artificial Intelligence*, 24, 7-83, 1984.

[21] Mortensen, N. H., Function Concepts for Machine Parts - Contribution to a Part Design Theory, *Proc. of ICED 99*, 2, 841-846, 1999.

[22] Borst, P., Akkermans, H., and Top, J., Engineering Ontologies, *Int'l Journal of Human-Computer Studies*, 46(2/3), 365-406, 1997.

[23] Horváth, I., Kuczogi, G.Y. and Vergeest, J. S. M., Development and Application of Design Concept, Ontologies for Contextual Conceptualization, In *Proc. of 1998 ASME*

Design Engineering Technical Conferences DETC, DETC98/CIE-5701, ASME, New York, 1998.

[24] Salustri, F. A., Ontological Commitments in Knowledge-based Design Software: A Progress Report, In *Proc. of The Third IFIP Working Group 5.2 Workshop on Knowledge Intensive CAD*, 31-51, 1998.

[25] Takeda, H., Veerkamp, P., Tomiyama, T. and Yoshikawa, H, Modeling Design Processes, *AI Magazine*, 11(4), 37-48, 1990.

[26] Invention Machine Co., *TechOptimizer*, http://www.invention-machine.com/prodserv/ techoptimizer.cfm, 2002.