# Document-based Design Process Knowledge Management for Knowledge Intensive Engineering

Yutaka Nomaguchi[1], Masaharu Yoshioka[2], Tetsuo Tomiyama[1]

[1]*Research into Artifact, Center for Engineering, The University of Tokyo*

[2]*Graduate School of Engineering, Hokkaido University*

**Abstract**: *In this paper, we propose a fundamental idea of a new CAD architecture to facilitate design knowledge management. This architecture is useful to develop an advanced design support system that encourages a designer to externalize his/her knowledge during a design process and facilitates sharing and reuse of such externalized design knowledge in later stages. We also describe an implementation of this idea called DDMS (Design Documentation Management System). DDMS works as a front-end to KIEF (Knowledge Intensive Engineering Framework) we have been developing. We also illustrate an example of machining tool design to demonstrate the features of DDMS..*

**Key words:** Computer aided design, Knowledge intensive engineering, Knowledge management, Design process, Design document

## 1. INTRODUCTION

The increasing concerns about product life cycle issues, such as environmental problems and product liability, make design so complex that a single designer cannot design without help of a computer that stores an intensive amount of knowledge. In addition, under today's competitive business environment, it is crucial to share and reuse existing design documents and past design experiences, including drawings of the product, referenced information to solve problems, alternatives suggested for specific requirements, calculation processes, evaluation methods of the product, and so on. For these reasons, knowledge management is increasingly becoming important in engineering design research.

One of the critical issues in knowledge management is knowledge acquisition. Documentation is an activity to record design rationale in order to acquire design information and knowledge, and to share and reuse them. However it is painful for designers to document all the design information during design.

Design rationale includes two types of design information (Suzuki *et al.* 1996). One is foreground information that is an explicit result of a design process, such as geometric drawings, CAD data, and product models. The other is background information that includes various kinds of information and knowledge used for generating such foreground information. Background information is significant to understand design rationale, but usually remains unconscious and after the design, it goes beyond the designer's memory. This makes the acquisition of design knowledge difficult. Besides, even if a design process is completely captured into documents, it is difficult for the designer to record a document appropriate for easy-retrieve, which makes design knowledge hard to share and reuse.

- This research aims at to develop a new methodology for managing design process knowledge and to build a knowledge management system as a new CAD system. This system should automatically generate design documents during design, so that the designer can externalize his/her knowledge into documents. Considering the nature of design knowledge stated above, we propose the following three requirements for the new CAD architecture; (i) documentation support to acquire knowledge, (ii) retrieval of relevant knowledge, and (iii) web-based knowledge publishing to share and reuse of the acquired design knowledge.

## 2. KNOWLEDGE INTENSIVE ENGINEERING FRAMEWORK

Knowledge intensive engineering (Tomiyama *et al.* 1996) is a new style of engineering to assist engineering activities in various product life cycle stages based on intensive use of the accumulated engineering knowledge. KIEF (Knowledge Intensive Engineering Framework) has been developed to support this concept (Sekiya *et al.* 1999). The core architecture of KIEF is depicted in Figure 1.

A metamodel is a symbolic network of concepts about physical phenomena and components of a design object (Kiriyama *et al.* 1992). It represents topological structure, attributes of a product, physical phenomena that would occur on the design object, and causality between phenomena. The metamodel is a snap shot of the design object at a certain moment. The metamodel also works as a workspace for designers in KIEF, and then design processes on KIEF are represented as sequences of operations to the metamodel.

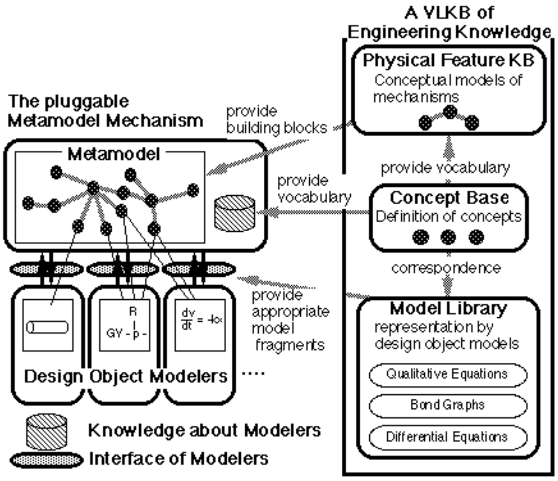


*Figure 1* The core architecture of KIEF

VLKB (Very Large-scaled Knowledge Base) accumulates various kinds of knowledge. The core of VLKB is the concept base (Ishii *et al.* 1995) that defines the ontology of physical concepts such as entities, relations, attributes, and physical phenomenon. These concepts form a vocabulary for the metamodel.

To integrate compartmentalized knowledge in multiple modelers, we employ a pluggable metamodel mechanism (Yoshioka *et al.* 1997) that defines correspondences between the concepts in the metamodel and the elements in the modelers at conceptual level. By doing so, the pluggable metamodel mechanism maintains relationships among the concepts used in the modelers and, together with integrated multiple modelers, offers the designer an effective design support environment.

Although KIEF itself was carefully designed and implemented with a user-friendly interface, still the system operation is difficult for many designers. This is primary due to the fact that integrated subsystems have their own interfaces and underlying concepts. One of the motivations of this research is to implement more unified KIEF's interface that incorporates the notion of design processes. (Ranta *et al.* 1995).

# 3. OVERVIEW OF DESIGN PROCESS

Before we propose our methodology to manage design process knowledge in KIEF, we overview our method to formalize design processes.

## 3.1 Design Information

First of all, we classify the design information that we treat in this research into the following four categories.

(1) *Information about the design object* is the information about the product itself, and includes such information as shape, dimension, and material. We further classify this type of information into the following four categories; (i) *models of the product* are engineering models; e.g., geometric models and finite element models; (ii) *attributes of the product* describe the parts and the structure of the product, such as length, weight, material; (iii) *values of the attributes* are concrete values of the attributes; and (iv) *constraints between the attributes* are mathematical expressions between the quantitative attribute values.

(2) *Information about the design process* is the information about the process to generate, modify, and delete the information about the design object. It is typically represented by operations on KIEF. Design tasks are interpretation of sequences of the operations from the viewpoint of design.

(3) *Information about the designer's intent* expresses the reasons why the designer generated, modified, or deleted the information about the design object. According to the cognitive argumentation model such as IBIS (Kunz *et al.* 1970), this information is represented by argument elements such as issues, positions, and so on. However since strict models here prevent the designer from flexibly writing rationale (Shah *et al.* 1998), we only use this as a guideline for the designer.

(4) *Information referred in the design* includes a variety of information, such as catalogues of the parts and design manuals in specific domains, with which the designer could perform the design.

## 3.2 Design Process Description

A design process is generally of trial-and-error nature. This means that it does not go straightforward from requirements to solutions, but goes back and forth between each of design alternative (Takeda *et al.* 1990). This is why we should consider a framework to describe trial-and-error design processes, although we can document design processes only sequentially.

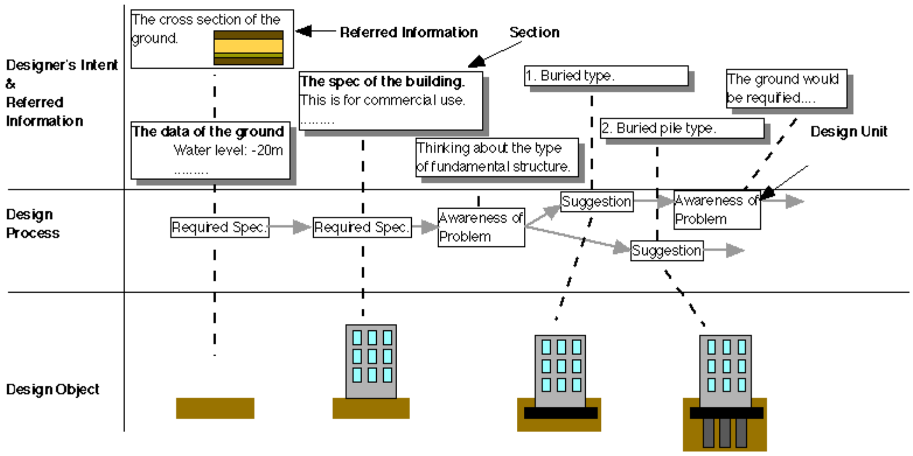


*Figure 2.* Design process description

A design unit represents a sequence of knowledge operations, such as *building models*, *suggestion of alternatives*, and *calculation,* and is generic and independent from design domains (Tijerino *et al.* 1993). It includes three categories of information, that is, *information about the product*, *information about the designer's intent*, and *information that is referred to during the design*. We then describe a design process as a network of these units.

In this research, a design unit is as a node in a document, called a *section*. Figure 2 shows an example design process description of a design case of civil engineering. The designer performs the design and simultaneously produces a document during the design. He/She designates the boundaries of a design unit, and defines the relationships between them. By doing so, the designer can produce information about the design process as a byproduct of the design (i.e., in addition to the product information) with helps from the system.

## 4. MANAGEMENT OF DESIGN PROCESS KNOWLEDGE

The goals of this research are to manage design process knowledge, and to share and reuse it. To manage design process knowledge, it is necessary to solve the following two issues; (i) acquiring the designer's knowledge, and (ii) retrieving relevant knowledge.

## 4.1 Documentation by Design

Acquiring the designer’s knowledge is one of the most critical issues in knowledge management. As we stated in Section 1, it is difficult for the designer to recall the design process and to guess about uncertain memories after the design has finished. Thus, we need a documentation method during design without bothering the designer from design activities. We propose here a method for a designer to produce a design document as a byproduct of design activities. We call this approach “documentation by design.” Recording the process of knowledge operations based-on the framework described in Section 2 allows reuse and reproduction of recorded documents easily. In the following, we describe the details of our approach.

### 4.1.1 Integrating A Documentation Tool and Engineering Tools

During design, designers have to perform a variety of activities, such as information gathering, various problem solving, drawing, and documentation. Recent computer technology can support these activities. Commercially available CAD systems support designers in the drawing activities. Analysis tools calculate complex formulae that describe a physical phenomenon occurring to a product, whereby reducing time for problem solving. E-mails and WWW tools allow geographically distributed designers to communicate easily and in a timely manner. Besides, word processors capable of taking information in a wide range of forms and formats are available to document design However, currently general documentation tools are not integrated with such tools. The designer has to go back and forth between them, if he/she wants to document design process during design. This is why these computer tools and a documentation tool should be integrated, and the documentation tool should be a unified interface of this integrated system to record design process as much as possible.

In this research, we employ KIEF as an integrated engineering tool environment. Our knowledge management system should include a documentation tool, which is the unified interface of KIEF. By *unified interface*, we mean the following two features.

- The designers allowed to access to each of integrated tools through the interface in a uniform manner.
- The interface can control operations on the integrated tools and record them automatically.

Although these features allow automatic documentation of design processes and information generated as a result of the operations performed with engineering tools, we do not restrict designer’s freedom to compose any sentences to encourage the designer to externalize his/her knowledge.

### 4.1.2 Hypertext Documentation

The form of design information varies from geometric data treated by a geometric modeler to such multi-media data referred to as figures and photographs. We need a framework to document these various kinds of information. In this research, we use a hypertext technique, such as HTML, to document design processes, which are described in Section 3.

## 4.2 Context-based Knowledge Retrieval

The most common technique for retrieving text documents is keyword matching. However, this often results in thousands of irrelevant search results and is practically inefficient. The Boolean query does not consider the relationship between keywords and the context of search, which are critical to perform appropriate retrieval.

To solve this problem, we introduce the concept of design contexts about relationships among different pieces of information and knowledge. We propose here to present a context as a set of words; e.g. *Context (Motor) ={large, torque}* means the context of *motor* is *large* and *torque*.

We model the context by extracting information from the metamodel of the product and texts in the design document, because the designer produces a document that contains product information presented by text and models during design. These kinds of information describe not only the product itself but also designer's intent through the design process, which consist of the design context. Figure 3 and the paragraph below show the algorithm to generate a context model from the metamodel and text information. Here, we mean by a *concept* the word already defined in the concept base of KIEF.

**Step 1:** Extracting the active, that is, currently composed sentences.

**Step 2:** Extracting active concepts from the metamodel. This feature allows the designer to exclude incorrect document knowledge, when he/she wants to retrieve document knowledge by using of a context model.

**Step 3:** Extracting sentences from these texts, and making a set of words for each of the sentences, e.g. a set of words {motor, generate, large, torque, available} is generated from a sentence "A motor that can generate large torque is available". We here employ morphological analysis to extract words and to remove meaningless words.

**Step 4:** Connecting entities in the metamodel and words that are determined as nouns. Ontology about entities accumulated in KIEF is used here. Suppose an entity "Stepping Motor" in the metamodel. Because "Stepping Motor" and "Motor" have a super-sub relation, "motor" in the group stated above is linked to "Stepping Motor" in the metamodel.

When the designer wants to retrieve documents similar to the currently performed design, documents that contain the same context would be ranked higher. Then the designer can select the most appropriate document.

The context model that represents prerequisites indexes of the knowledge in the document. Thus, we can grade similarity between context models to retrieve knowledge relevant to the currently performed design document. This similarity is graded by the number of matched concepts of the context model and by the number of the matched words linked to the matched entities.

```
ContextModelingFor: currentDU

|
currentDU    " currently active design unit; Oe: Ontology of entity"
activeSentences    " Set of sentences; activeConcepts: Set of concepts in metamodel"
activeEntities    " Set of entities"
answer    " Set of context"
|

        answer := Collection new.
        activeSentences := currentDU getActiveSentences.                    " Step 1 "
        activeConcepts := currentDU getActiveConcepts                       " Step 2 "
        activeEntities := activeConcepts selectEntities.
        sentences do:
        [:sentence|
                activeWords := sentence morphologicalAnalysis.              " Step 3 "
                activeEntities do:
                [:entity|
                        |entitySynonyms|
                        entitySynonyms:= Oe getSynonymsOf: entity.
                        (etitySynonyms intersection: activeWords) isEmpty) ifTrue:
                        [
                                answer add: (entity -> activeWords).        " Step 4 "
                        ]
                ]
        ]
```

*Figure 3* Algorithm of generating context model

## 5. IMPLEMENTATION OF DDMS

Based on the discussions above, we have developed a Design Documentation Management System (DDMS). The system was developed in VisualWorks 3.0(a registered trademark of Cincom systems, Inc.), a version of Smalltalk-80, and VisualWave that is an add-on tool for development/employment of web applications, so that documents composed with DDMS can be published to geographically distributed designers who wish to access to knowledge of KIEF.

DDMS contains a document editor running on top of KIEF. The designer can use modelers plugged into the metamodel system of KIEF; such as the FBS modeler (Umeda *et al.* 1996) to support functional design, a qualitative reasoning system that analyse behaviour of a product qualitatively, and other analysis modelers. These modelers can uniformly be operated from DDMS that can record operations to them. Figure 4 shows the architecture of the system.

## 5.1 The Functions of DDMS

DDMS's fundamental function is a HTML (Hyper Text Markup Language) text editor, which allows the designer to compose texts in HTML at anytime. In addition, DDMS has the following functions.

1) Recording design process.

Knowledge operations on KIEF during the design are automatically converted into a natural language by a document converter of DDMS, and recorded on the document preserving step-by-step and back-and-forth natures of design processes with a hypertext technique.

2) Design process management

We employ a design process management mechanism based on ATMS (Assumption-based Truth Maintenance System) (de Kleer 1986) embedded in the metamodel mechanism of KIEF. The designer can use this feature on the document editor. When the designer switches a section to another in order to work on an alternative, the metamodel mechanism justifies only information operated in the new section and its ancestral ones.

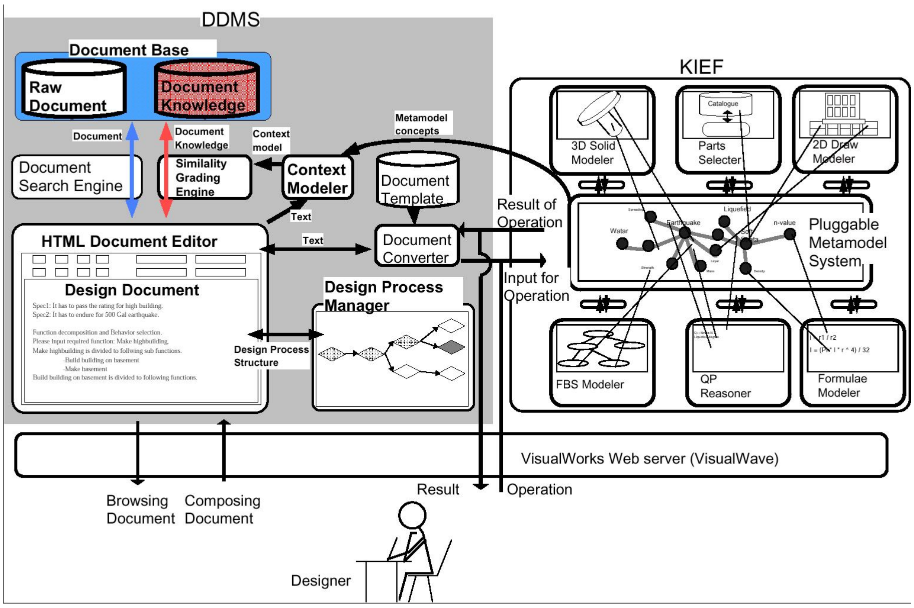


*Figure 4.* The architecture of DDMS

3) Document knowledge base.

DDMS has a knowledge base to store document knowledge. The designer can retrieve documents with the retrieval method described in Section 3.

4) Recording history of information reference.

DDMS records information referred to during design, such as files, web pages, and models generated in the design process and links to them in the design document as hyperlinks of HTML for future reuse.

5) Documenting the designer’s intent.

DDMS supports the designer to document the designer’s intent in two ways. One is to prompt the designer to enter his/her intent in a natural language every time the designer performs an operation. The other is to list possible contexts of the document knowledge for the designer to select one of them as his/her intent, when DDMS retrieves document knowledge.

6) Knowledge publishing on web

Documents produced by DDMS are coded in HTML, so that they can be published through a commercially available Web browser.

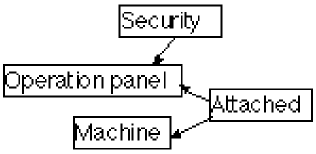
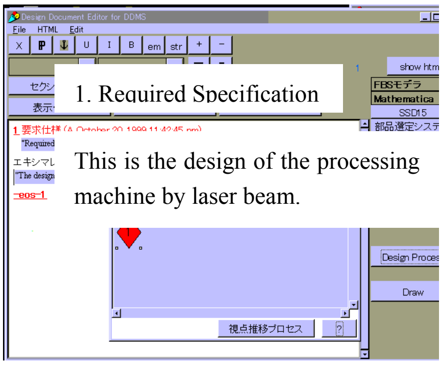
| Contents of the document knowledge | Context model |
|---|---|
| **The position of a operation panel from the view point of security** :<br>An operation panel should be at the position where a worker operation without opening a save cover of a machine |  |

*Figure 5* Example of document knowledge

## 5.2 Design Session

We illustrate an example of machining tool design and demonstrate the power of the system. First, we collected more than 40 chunks of document knowledge from the design documents produced during an actual design activity. In addition, we could collect drawings of the products, referred information to solve a problem, alternatives suggested for specific requirements, calculation processes, evaluation methods of the product, and so on. Figure 5 shows an example of document knowledge.

Hereafter, we trace the design process to explain DDMS. First, the designer writes initial requirements for the product on a document as a memorandum. In this example, the designer determines that the design should use a laser beam to process a work (Figure 6). Then the designer sketches a drawing model of the product with a 2D sketching modeler connected to the pluggable metamodel mechanism. On this sketch, the designer can designate concepts that correspond to each part in the sketch. This model will be fed into the metamodel mechanism. This model is saved and automatically hyperlinked on the document (Figure 7). The designer wants to retrieve documents to suggest solution candidates about a sketched product. DDMS then generates a context model from the metamodel and text written on the document (Figure 8). This model means that the context of currently performed design is, e.g., Context(Jig) = {suggest, process, work, laser, beam}.



1. Required Specification

1 要求仕様 (A October 20 1999 11:42:45 pm)

"Required

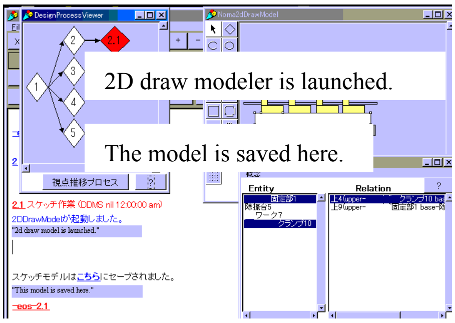
エキシマレ

"The design

This is the design of the processing machine by laser beam.

−eos−1

*Figure 6* Writing specification for the product



2D draw modeler is launched.

The model is saved here.

2.1 スケッチ作業 (DDMS nil 12:00:00 am)

2DDrawModelが起動しました。

"2d draw model is launched."

スケッチモデルはこちらにセーブされました。

"This model is saved here."

−eos−2.1

| Entity | Relation |
|---|---|
| 固定部1 | 上4(upper− クランプ10 bas |
| 除振台5 | 上9(upper− 固定部1 base−除 |
| ワーク7 | |
| クランプ10 | |

*Figure 7* Sketching the product

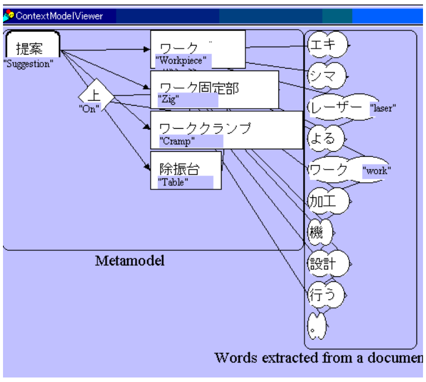
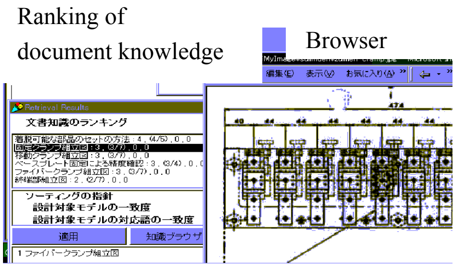


*Figure 8* Context model


Ranking of document knowledge

Browser

MyImage¥sumen¥zumen cramp.jpg Microsoft In
編集(E) 表示(V) お気に入り(A) »
Retrieval Results
適用
知識ブラウザ
474
40
44

文書知識のランキング

着脱可能な部品のセットの方法：4，(4/5)，0，0
固定クランプ組立図：3，(3/7)，0，0
移動クランプ組立図：3，(3/7)，0，0
ベースプレート固定による精度確認：3，(3/4)，0，0
ファイバークランプ組立図：3，(3/7)，0，0
終端部組立図：2，(2/7)，0，0

ソーティングの指針
設計対象モデルの一致度
設計対象モデルの対応語の一致度

1 ファイバークランプ組立図

*Figure 9* Browsing ranked document knowledge

DDMS retrieves document knowledge from the document base, and ranked them (Figure 9). Document knowledge about “jig of a laser-beam processing machine” is ranked higher than document knowledge indexed “terminator of laser beam,” “a safety position of operation panel of processing machine” and so on. The designer can browse them on a Web browser and select highly ranked ones. Selected document knowledge is applied to the currently composed document.

## 6. DISCUSSION

DDMS employs the pluggable metamodel mechanism supported by the knowledge operation model for semi-automatic design documentation and context-based knowledge retrieval. We here discuss DDMS's capabilities to acquire design knowledge by semi-automatic documentation.

*Table 1* Comparison of the number of information

| | Words | Numerical Values | Figures and Photographs | Engineering models |
|---|---|---|---|---|
| Manually composed document | 350 | 21 | 4 | 0 |
| Document produced by DDMS | 1200(1000, 200) | 96 (18, 78) | 4 (4, 0) | 7 (3, 4) |

Table 1 shows the result of this comparison. The numbers in brackets show the breakdown of information recorded by the system (left side), and that created manually by the designer (right side). This result shows that documentation by DDMS can capture much more information than manual documentation. The reasons for this include; (i) DDMS captures information of design process, not only design results, (ii) DDMS keeps information about discarded alternatives that are usually forgotten or neglected in manual documentation, (iii) DDMS hyperlinks referenced information and used engineering models on the document, which is impossible with manual documentation, and (iv) DDMS prompts the designer to write his/her comments at the time when he/she performs a design activity.

These features facilitate to capture design process information to understand the product, although we need to evaluate the overhead of this approach that might take much more time to design than usual.

## 7. RELATED WORKS

Recently, research efforts about knowledge management are actively reported (e.g. Dieng 2000). These include web-based implementation, XML-based data format, and ontological databases to arrange stored documents. While these efforts assume the existence of already-acquired knowledge to manage, we focus on acquisition of knowledge, which is the crucial issue for knowledge management.

From the viewpoint of design research, we can find out several reports about capturing design rationale. Garcia classifies the attempts to capture design rationale into three categories, i.e., model-based rationale, argumentation-based rationale, and action-based rationale (Garcia *et al.* 1992). These methods have both pros and cons.

Our approach combined these three methods to capture design rationales taking advantages of each of them. First, it is model-based, because we use the metamodel concept to reflect the designer's mental model. Second, it is argumentation-based, because DDMS allows geographically distributed designers to interact each other by documenting design. Third, it is action-based, because DDMS records the designer's actions during design..

## 8. CONCLUSIONS

We proposed a methodology to management of design process knowledge, and based on this, we have developed a design document management system called DDMS. DDMS works as a front-end of KIEF and capture knowledge operations on KIEF. This feature facilitates us to capture design knowledge during design, and to retrieve knowledge to share and reuse it in the future design

## Acknowledgment

This research was supported by the international collaborative research project, GNOSIS, in the Intelligent Manufacturing Systems (IMS) research program. The example of engineering design of machine tool was developed under the collaboration with Sumitomo Electric Industries, LTD (SEI). We would like to thank Dr. Masayoshi Fuse, Mr. Hisashi Togawa, Mr. Yutaka Nishide, Mr. Shuuichi Miyake and Mr. Shoichiro Fujisaki of SEI. This work has also been partially financed by JSPS "Modelling of Synthesis" project (JSPS-RFTF96P00701).

## References

de Kleer, J. (1986): "An assumption-based TMS," Artificial Intelligence, Vol. 28, pp. 127-162.

Dieng, R. (2000) "Knowledge Management and the Internet", IEEE Intelligent Systems, Vol. 15, Issue. 3, pp. 14-17.

Garcia, A.C.B. and Howard, H.C. (1992) "Acquiring Design Knowledge through Design Decision Justification," AIEDAM, Vol. 6, No. 1, pp. 59-71.

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.

Kiriyama, T., Tomiyama, T. and Yoshikawa, H. (1992) "Qualitative Reasoning in Conceptual Design with Physical Features," in B. Faltings and P. Struss (eds.): *Recent*

*Advances in Qualitative Physics*, The MIT Press, Cambridge, MA, USA, (1992), pp. 375-386.

Kunz, W., and Rittel, H., (1970): “Issues as elements of information systems,” Working Paper No. 131, Berkeley, University of California, Berkeley, Institute of Urban and Regional Development.

Ranta, M., Mäntylä, M., Umeda, Y., and Tomiyama, T. (1996) “Integration of Functional and Feature-Based Product Modeling The IMS/GNOSIS Experience,” Computer-Aided Design, Vol. 28, No. 5, pp. 371-381.

Sekiya, T., Yoshioka, M., and Tomiyama, T. (1999) “The Development of Knowledge Integrated Engineering Framework Based on Ontology,” Journal of Japanese Society for Artificial Intelligence, Vol. 14, No. 6, pp. 1051-1060.

Shah, J., Rangaswamy, S., Qureshi, S., Urban, S. (1998). Design History System: Data Models & Prototype Implementation, In Finger, S., Tomiyama., and Mäntylä, M. (eds.): *Knowledge Intensive CAD Volume 3,* pp. 91-114. Chapman & Hall

Suzuki, H., and Kimura, F. (1996) “Modeling Information in Design Background for Product Development Support”, Annals of the CIRP, pp. 141-144.

Takeda, H., Veerkamp, P., Tomiyama, T., and Yoshikawa, H., (1990): “Modeling Design Process,” AI Magazine, Vol. 11, No. 4, pp. 37-48.

Tijerino, Y, -A., Ikeda, M., Kitahashi, T. and Mizoguchi, R,. (1993). “A Methodology for Building Expert Systems Based on Task Ontology and Reuse of Knowledge –Underlying Philosophy of a Task Analysis Interview System Multis”, Journal of Japanese Society for Artificial Intelligence, Vol. 8, No. 4, pp. 476-487 (in Japanese)

Tomiyama, T., Umeda, Y., Ishii, M., Yoshioka, M., and Kiriyama, T. (1996) ” Knowledge systematization for a knowledge intensive engineering framework,” In T. Tomiyama, M. Mäntylä, and S. Finger (eds.): Knowledge Intensive CAD-1, Chapman & Hall, London, pp. 300-314.

Umeda, Y., Ishii, M., Yoshioka, M. and. Shimomura, Y. and Tomiyama, T. (1996) “Supporting Conceptual Design Based on the Function-Behavior-State Modeler,” Artificial Intelligence for Engineering Design, Analysis and Manufacturing, Vol. 10, No. 4, pp. 275-288.

Yoshioka, M., and Tomiyama, T. (1997) “Pluggable Metamodel Mechanism: A Framework of an Integrated Design Object Modelling Environment,” in A. Bradshaw and J. Counsel (eds.), Computer Aided Conceptual Design ’97, Proceedings of the 1997 Lancaster International Workshop on Engineering Design CACD ’97, Lancaster University, Lancaster, UK, pp. 57-70.