We are IntechOpen, the world's leading publisher of Open Access books
Built by scientists, for scientists

14

Study on Product Knowledge Management forProduct Development

Chunli Yang[1], Hao Li[2] and Ming Yu[3]
[1]China Center for Information Industry Development, Beijing, 100846, China
[2]Research Center for Medical and Health Management, School of Economics and management, Tsinghua University Beijing, 100084, China
[3]Department of Industry Engineering, Tsinghua University, Beijing, China

1. Introduction

The goal of engineering product development in today's industry is to provide products meeting individual requirements at the lowest cost, the best quality and the shortest time. Abundant design knowledge is needed, and cases and designers' experiences should be utilized at most as possible. In addition, product development is becoming more often done collaboratively, by geographically and temporally distributed design teams, which means a single designer or design team can no longer manage the complete product development effort. Therefore it is necessary to collect and manage the design knowledge to support share and pass of them among designers. In some sense, quick capture and effective use of design knowledge are essential for successful product development.
The modern manufacturing environment and the new product development paradigms provide more chances with enterprises and customers to cooperate among different enterprises, different departments of a firm, enterprises and their customers, etc. Designers are no longer merely exchanging geometric data, but more general knowledge about design and design process, including specifications, design rules, constraints, rationale, etc (Simon Szykman, 2000). Product development is becoming increasingly knowledge intensive and collaborative. In this situation, the need for an integrated knowledge resource environment to support the representation, capture, share, and reuse of design knowledge among distributed designers becomes more critical.
A great deal of technical data and information including experience generated from product development is one of the most important resources of product knowledge. It is necessary to use knowledge based information management methods and technologies, which can dig and capture product knowledge from those resources supporting product development. The engineering design community has been developing new classes of tools to support product data management (PDM), which are making progress toward the next generation of engineering design support tools. However, these systems have been focusing primarily on database-related issues and do not place a primary emphasis on information models for artifact representation (Simon Szykman & Ram D. Sriram ,2000). Furthermore, although these systems can represent non-geometric information—for example, about design process,

manufacturing process, and bills of materials — representation of the artifacts is still generally limited to geometry. For example, PDM techniques focus on product data management but little product knowledge, and they are limited to capture, organization and transfer of product knowledge (Ni Yihua, Yang Jiangxin, Gu Xinjian,et,al, 2003). Moreover, they are unable to elicit knowledge from lots of product data and cannot satisfy the requirements of knowledge management in product development. In such cases, the need for building a kmowlege management system to support PLM (product lifecycle management) becomes more critical. Such kmowlege management system can not only represent the knowledge of product and product development processes, but also support firms to quickly identify, capture, store and transfer the knowledge, on which a better and more effective mechanism of knowledge accumulation and management is formed.

In response, the main purpose of this paper is to study product knowledge management methodologies, build an integrated knowledge management framework for decision-making, and develop a software prototype to support quick capture and reuse of knowledge during product development. The remaining part of this paper consists of five main sections: 2. Related research work; 3. Product knowledge management system (PKMS) framework; 4. Semantic object network model; 5. Product knowledge management process; 6. Design repository; 7. Implementation and application of PKMS; 8.Conclusions. Following on from a brief literature review to construct a PKMS research framework, a great portion of this paper focuses on the product knowledge management process, along with several illustrations of software prototype. The paper ends with concluding remarks.

2. Related research work

Product development is complex system engineering. It involves in representation, capture, and reuse of product knowledge. Recently, researches on knowledge representation, acquisition and management are emphasized increasingly.

2.1 Product knowledge representation

The knowledge representation is the core issue in AI and many representational methods such as logic and predicate mode, procedure mode, production system, semantic network, framework, knowledge unit, case base, and object orientation, etc. have been reported in AI to meet the requirements for the specific problems. Production system, semantic network, framework, case base, object orientation, and graph, etc. have been used to represent product knowledge in mechanical engineering, in which object orientation, rule-based, and hybrid representation schemes are popular.

2.1.1 Object orientation representation

X.F.Zha provided an integrated object model to represent product knowledge and data, which supports calculating and reasoning work in assembly oriented design processes (X.F. Zha,2001). The integrated object model employed an object orientation scheme and the knowledge P/T net formalisms to form a hierarchy of function-behaviour, structure, geometry and feature. Such model was used as a uniform description of assembly modelling, design and planning.

The SHARED object model is presented to realize conceptual design (S R Gorti &A Gupta,1998). It clearly defines relationships among objects. Object-oriented technology makes it possible to naturally decompose design process and hierarchically represent design knowledge. S R. Gorti and etc (Simon Szykman &Ram D. Sriram ,2000). presented a flexibly knowledge representation model based on SHARED. It further extends object-oriented design technology, and represents knowledge of product and design process by combining products and their design processes according to the hierarchical structures. The model encapsulates design rationale by using structured knowledge code. Artifacts are defined as the composition of three kinds of objects: function, form and behavior. Form represents physical performance. Behavior represents consequence of operations.

A product modelling language is developed (Nonaka,1991). It defined products as object sets and relations. This product modeling language includes data language (DL) and design representation language (DPL). DL independent of any engineering environment is defined as basic framework of general object template and data structure. DPL provides methods of setting product model by combining DL with engineering environment. The method supports complex pattern matching algorithm based on graph and provides a neutral language to capture and exchange product information. It uses effective methods to store and reuse knowledge (Simon Szykman,2000).

Oliva R. Liu Sheng and Chih-Ping Wei proposed a synthesized object-oriented entity-relationship (SOOER) model to represent the knowledge and the imbedded data semantics involved in coupled knowledge-base/database systems (Sanja Vranes,1999). The SOOER model represents the structural knowledge using object classes and their relationships and encapsulates the general procedural, the heuristic and the control knowledge within the object classes involves.

XB, liU and Chunli YANG presented a product knowledge model which is built with object modelling techniques (XB, LIU & Chunli YANG,2003) . In order to easily realize knowledge management, the object model is mapped to the Relation Database.

2.1.2 Graph-based representation

In order to directly capture the relationships among design attributes (geometry, topology, features) and symbolic data representing other critical engineering and manufacturing data (tolerances, process plans, etc.), W.C. Regli presented a graphical structure called as a design signature, i.e. a hyper-graph structure H(V,E) with labeled edges is used to represent the a mechatronic design and its design attributes (W.C. Reglil, V.A. Cicirello,2000). All vertices representing design attributes are connected to the vertices representing the entities in the boundary model that attributes refer to. Such representation method can facilitate retrieval of models, design rules, constraints, cases, assembly dada, and experiences and identifying those products with similar structures or functions, which helps designers to better perform the variant designs based on cases.

Yu Junhe used structural hypergraph to describe the hierarchical structure of sets in a product family structure model. The evolving hypergraph networks represent the information on design processes, which can trace the historical information and facilitate retrival and reuse of the product information (YU JunHe, QI Guoning,WU Zhaotong, 2003).

Knowledge representation based on graph, such as knowledge map, concept map, hyper-graph and so on, belongs to the semantic network category, which has many characteristics, e.g. its structure is simple, easily readable, can truly describes the semantics of the natural language and has more precise mathematics bases. They will be used in many domains, especially in the knowledge representations based on Web, the topic map is the development tendency (Zhang lei &Hou dianshe, 1998).

2.1.3 STEP-based representation
Recently, as more CAD companies incorporate the ability to import and export STEP data, the use of the STEP AP 203 as a neutral format will leverage research efforts and maximize interoperability with existing CAD software used in industry. STEP provides a standard modeling method. Yang zijiang proposed a STEP-based organization, transfer and exchange mechanism for product information model (Yang Zijiang, Yang Xiaohu, Li Shanping, Dong Jianxiang, 2001). The NIST design repository also used STEP as representing the geometry information.

2.1.4 Generalized representation
Data, information and knowledge have some inner relationships. In order to effectively support product development, many researchers proposed that data, information and knowledge should represnt uniformily, thus a generalized knowledge representaion model was presented, which was organizaed, stored and managed by using database management systems. E.g. Cao Jian thought the data and knowledge invovled in the product development processes as the product knowledge (Cao jian, Zhang Youliang, Huang Shuangxi, Zhou Xinguang, 1999). Qi Yuansheng defined the generalized design knowedge as design information which was used to make decisions (Qi Yuansheng,Peng Hua, Guo Shunsheng, Yang Mingzhong, 2002). The generalized design informaion includes not only the formal design information and engineering information , such as industry documents, formulae, CAD data, tolerances, solutions,etc. , but also market information, forecasting information and some decision-making information. He used multi knowledge representaions including object-oriented, ontology, XML-based methods. The BEST's knowledge-representation language Prolog/Rex (Vranes^E and StanojevicÂ , 1994) attempts to combine rules, frames and logical expressions under a unique formalism.
All of the above mentioned knowledge representaion methods combine the product development and its charactristics. In addtion, they discribe design, design process and the used knowledge accroding bo the given domains. Their goals are to fast develop, design and manufacture good products. However, how to represent the design historical information, design rationale, expriences and other tacit knowledge needs to deep research. In addition, product knowledge has complicated semantics and heterogeneous constrains. Therefore product knowledge representation model should define and represent these semantics and constrains in order to subsequently share, transfer, and reuse product knowledge.

2.2 Product Knowledge Acquisition
Engineering product development is a knowledge intensive and creative action. The whole product development process will access and capture quite lots of knowledge about design and design process. However, the knowledge acquisition issues also become the bottleneck during the product development processes for lack of effective knowledge capture methods and techniques, e.g. only those salted "old designers" can do some product development activities well. Recently, in order to alleviate the knowledge acquisition issues, researches on knowledge acquisition are emphasized increasingly.

2.2.1 Design catalog

Design catalogue is used to capture and store engineering product design knowledge(Zhang han, Zhang yongqin,1999). Ontology is also employed to aid acquire product knowledge, since ontology provides an explicit scheme of structuring knowledge content according to coherent concepts. Designers can easily find the needed knowledge though ontology.

2.2.2 Ontologies

Soininen T. (Soininen T. ,2000)presented configuration ontology for representing knowledge on the component types in product configuration design. Recently most of design and manufacturing knowledge is stored in digital form in computer, therefore knowledge discover and data dig methods and tools are contributed to knowledge acquisition(Zhu cheng, Cao wenze,2002).

2.2.3 Multi knowledge-capturing schema

Generally, a multi knowledge-capturing schema is needed as product design requires many kinds of knowledge and one kind of knowledge acquisition method cannot capture all kinds of knowledge. Many kinds of knowledge acquisition method are integrated with work in product design process, which satisfies its requirements of knowledge (Zhang shu, Li aiping ,1999).

2.2.4 Web-based knowledge capture

In addition, web-based computer aided design technologies are becoming new product development methods driven by customer requirements, which needs to capture the relative knowledge through Internet/Intranet environment (Lai chaoan, Sun yanming, 2002). Web-based information platform provides abundant information resource for product innovation development. However, the traditional knowledge acquisition methods have not met the requirements of knowledge capturing, disposal, and use in the Internet/Intranet environment. So many researchers begin to study how to quickly acquire knowledge by Internet technologies.

A great deal of technical data and information including experience generated from product development exists in the form of files, communications, meeting notes and Emails, in which, design semantic information, such as design intent, design rationale, etc. is regarded as the important knowledge resource and the basis of new product development and the past product improvement. There are abundant product knowledge resource, but designers usually do not know where to find the right design knowledge, how to understand the original design purpose, how to reuse design methods, and so on. The main reason is that these technical data and information cannot be effectively organized, stored, and accumulated due to lack of methods and tools of knowledge acquisition, which leads to many efforts spent on retrieval.

3. Product knowledge management framework

To address those issues described above, we proposed the product knowledge management framework. It contains five main components: design repository (DR), OLAP, knowledge reduction, Case Based Reasoning (CBR), and machine learning, as shown in Fig.1.

3.1 Design Repository

DR (Design Repository) takes the role as a knowledge base where contains all the information and knowledge related to products. Furthermore, it is responsible for collecting, extracting, converting, cleaning, aggregation, and indexing the information and knowledge from various kinds of information and knowledge sources.

Knowledge in DR is organized into a subject-oriented and multi-dimensional knowledge model showed in Fig 2. Product design knowledge has five subjects: domain knowledge, product knowledge, design cases, knowledge of customer and market, and design methods. The subjects can be divided into sub-subjects according to requirements of product development.

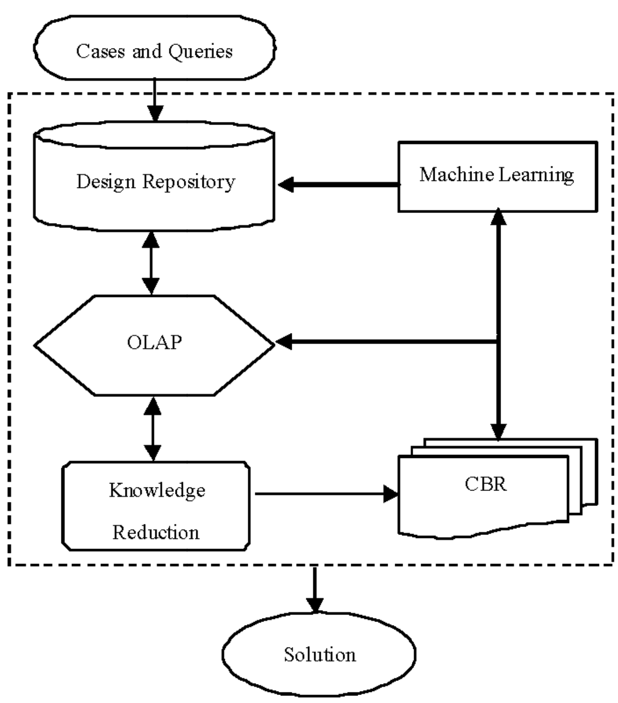


Fig. 1. Framework of Product Knowledge Management

3.2 OLAP technique

The purpose of OLAP in the framework is to facilitate timely access of the knowledge in DR though the application to roll up, drill down, slice, dice, and etc. knowledge to obtain further information. CBR takes advantages of OLAP techniques of its flexible and timely manipulate of data and its special functions, which may speed up attributes matching and cases searching.

3.3 Knowledge reduction

To retrieve a similar case, CBR attempts to match the similar condition between the new case and existing cases in a case-based repository. In traditional approaches, Nearest-Neighbor Retrieval (NNR) approach is used to compute the similarity between stored cases and the new case, based on features and variant weights of the cases. The NNR process examines the similarity for all cases in the case-based repository, which is time-consuming and inefficient when the scale of the case becomes large. In addition, traditional CBR is deficient for case-reduction of information-preserving data, which results in redundant similarity testing and the computation time consuming (Huang C. C, Tseng T L.,2004). In order to avoid them, it is necessary to integrate rough set approach into CBR, in which knowledge reduction method can induce rules based on previous data in the cases to reduce the number of similarity testing and the computation time consuming (Huang C. C, Tseng T L.,2004). In order to avoid them, it is necessary to integrate rough set approach into CBR, in which knowledge reduction method can induce rules based on previous data in the cases to reduce the number of similarity testing cases and their attributes from OLAP module.

3.4 CBR

CBR model in this paper is extended to the R4 model, presented by Aamodt and Plaza, for satisfying the requirements of product agile development, as shown in Fig.2.
Case retrieval: Once a new inquiry has been input CBR module, it will be sent to OLAP which performs search and match to find the attributes that are similar to the inquiry's attributes in DR. OLAP will retrieve all the similar cases by drilling, slicing, and dicing the DR. Then knowledge reduction module reduces all the similar cases and their attributes so that NNR could avoid test redundant similarity cases and their attributes. After NNR examines the similarity for the reduced similar cases, finally CBR get the most similar case. Case reuse: The retrieved case is combined with the new case—through REUSE — into a solved case, i.e. a proposed solution to the initial problem. Case revises: The new case is revised in structure or dimension according to the configuration rules and constraints. Then the revised case is tested for whether it meets the requirements, e.g. by being applied to the real environment or evaluated by customers, and repaired if failed. Case retains: Useful experience is retained for future reuse, and the DR is updated by a new case, or by modification of some existing cases.

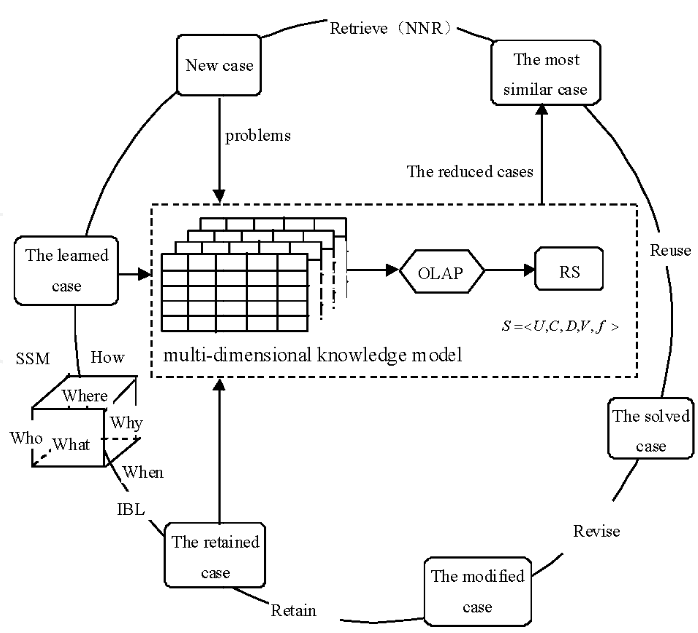


Fig. 2. CBR process model and machine learning process

3.5 Machine learning

Machine learning module is responsible for extracting process information and knowledge from CBR cycle and builds Solution Subject Model (SSM). SSM is stored as a kind of design rationale in DR. SSM includes all the background knowledge such as deliberating, reasoning, trade-off and decision-making in the CBR cycle of an artifact—information that can be valuable, even critical to various designers who deal with the artifact. Therefore, SSM contains a description of why it is revised the way it is, and more detail, it also includes how it is revised, who revised it, when it is revised, which part is revised, etc. SSM can be used as a basis to record the history of configuration and variant design process; to modify and maintain existing designs, or design similar artifacts.

4. Semantic object knowledge representation model

Knowledge in product customization design process is diverse. Design objects change dynamically with the change of the design process, and knowledge used during the whole process changes, too, even is iteratively used. Thus, product knowledge representation model should be constructed according to the features of product design and the process.

4.1 Semantic object network

From above analysis, semantic object network is used to represent knowledge on product customization design. Semantic object network is a kind of Semantic Networks (SNs), in which nodes are semantic objects. Semantic object networks link all information, documents and other descriptive knowledge related to the final products. Therefore, Semantic object networks are convenient for product structure and configuration management.

Semantic Object (SO) is a naming collection of attributes that sufficiently describes a distinct identity (Huang Wei, 2001). Attributes are described by attribute types, cardinality and domains. SO is divided into simple objects and composite objects according to attribute characters (Li SiQiang ,1999).

The nodes in Semantic object network represent SO, and directed arcs represent relations among nodes including Generalization Association (G), Aggregation Association (A), Composition Association (C), etc. AKO (a kind of), APO (a part of), and ACO (a composition of) respectively represents the G association, the A association, and the C association.

4.2 Domain knowledge semantic object network

Domain knowledge updates slowly. Designers usually use the definite domain knowledge for the specific tasks during product design process. According to this distinct character, domain knowledge semantic object networks are built, in which, all the semantic objects are basic objects and simple objects, as shown in Fig. 3.

Most of domain knowledge embodies in documents. Besides, some of domain knowledge embodies in program files, material, calculation methods, etc. Therefore, the semantic objects in domain knowledge semantic object networks are program objects, document objects, material objects, and calculation method objects. The document objects have three types: structured document objects, non-structured document objects and drawing document objects.

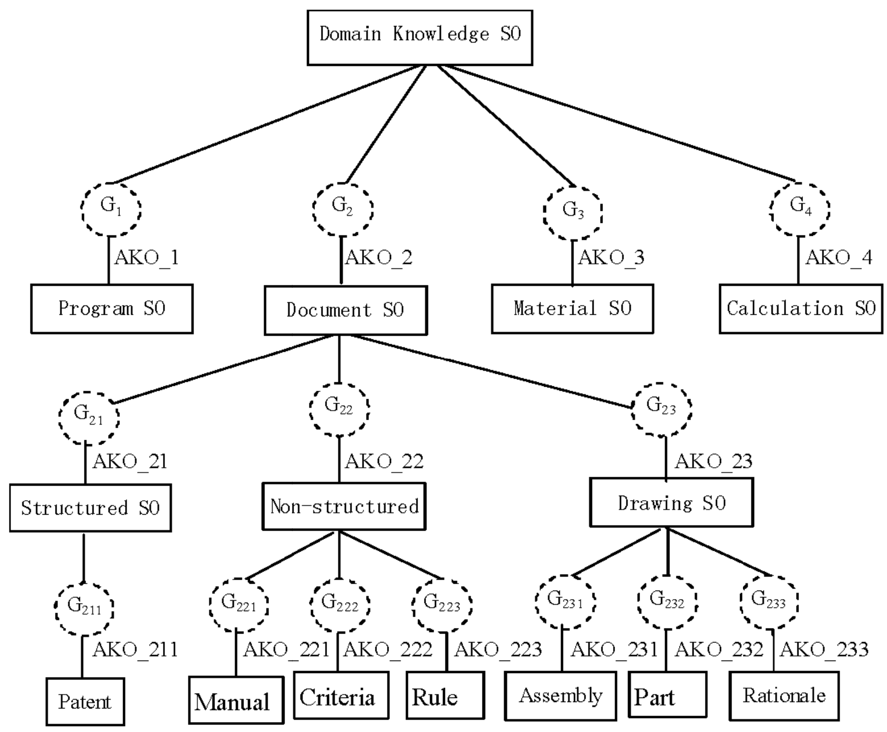


Fig. 3. Domain Semantic Object Networks

4.3 Product function semantic object networks

Product Function Semantic Object Networks (PFSONs) represent knowledge about product concept including customer requirements, product functional features, behavior knowledge, and design rationale. So, PFSONs are composed of four sub SNs: function, behavior, requirement, and design rationale, as is shown in Fig. 4.

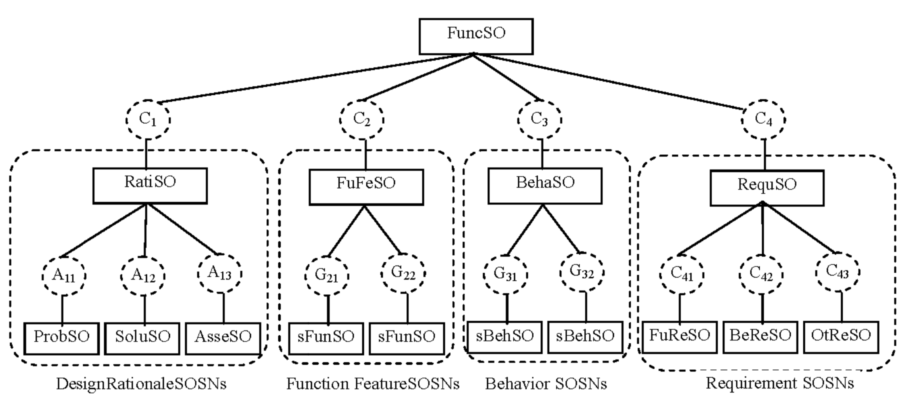


Fig. 4. Product function semantic object networks

Requirement Semantic Object (RequSO) is the collection of customers' requirements including structure, function, appearance, and other requirements. So RequSO includes structure requirement semantic object (StReSO), function requirement semantic object (FuReSO), appearance requirement semantic object (ApReSO), and other requirement semantic object (OtReSO). Function Feature Semantic Object (FuFeSO) represents knowledge of product function and engineering attributes. FuFeSO can be divided into sub FuFeSO (sFunSO). Behavior Semantic Object (BehaSO) specifies the response of an artifact to input conditions or behavioral states (or both). BehaSO can be divided into sub BehaSO (sBehaSO). Design rationale is a kind of process knowledge. It can encompass the documentation of the active processes of reasoning and decision making that led to the artifact design — including the justification for design decisions, records of design alternatives considered and tradeoffs evaluated, and details of the argumentation and communication processes associated with design work. So, design rationale semantic object (RatiSO) includes problem semantic object (ProbSO), solution semantic object (SoluSO), and evaluation semantic object (EvalSO).

4.4 Product configuration management semantic object networks

Configuration rule semantic objects (CRSO) are composed of valid configuration rule objects (ValidCRSO), variable configuration rule objects (VariableCRSO) and version configuration rule objects (VersionCRSO). Configuration constraint semantic objects (CCSO) are divided into variablesolving constraint objects (VSCSO), variable-judging constraint objects (VJCSO), logic constraint objects (LCSO). Product configuration management semantic objects (PCMSO) are composed of CRSO, CCSO and version semantic objects (VSO). Fig.5 presents the structure of PCMSO. Product structure management and configuration design are realized through versions management mechanism according to configuration rules and constraints.

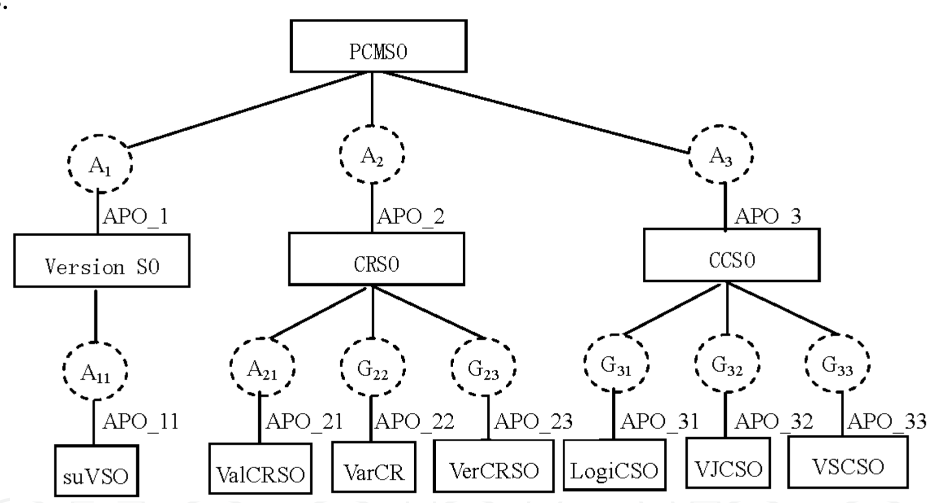


Fig. 5. Product configuration management semantic object networks

5. Product knowledge management process

The purpose of product knowledge management is to provide a mechanism to capture, organize, store, and reuse knowledge for product agile design. It aims to help designers timely find information and knowledge that is needed and enable the maximum reuse of existing knowledge resource in the development of a new product.

5.1 Knowledge analytical model

Product agile development is derived by customer needs. Thus the start point of it is to transform customer needs into engineering design attributes. These engineering design attributes, as the input conditions, will be input into CBR. The analytical model of product agile customization design is showed in Fig. 6.

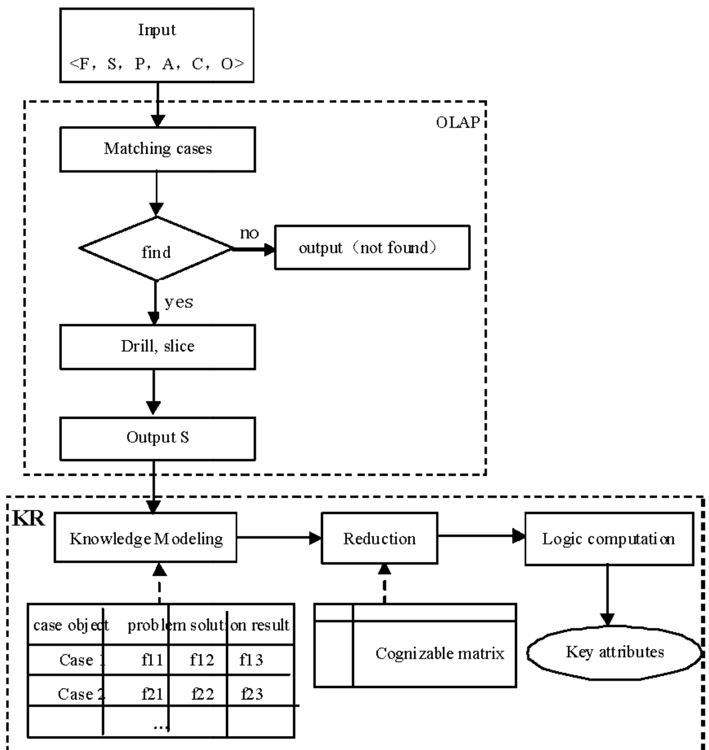


Fig. 6. Knowledge analytical

Firstly, input conditions, i.e. engineering design attributes, are input into CBR for retrieval the similar cases. The engineering design attributes are represented as an input model: I= (F, S, P, A, C, O). Where they separately denote function, structure, performance, appearance, cost, and other attributes. Then I will be sent to OLAP which performs search and match to find the model attributes that are similar to I in DR. OLAP would use multi-dimensional approach to obtain a specific data by going into proper dimension. OLAP will also drill down further for the differences that are found between the similar cases and I in order to get the best solution. Thirdly, knowledge reduction module reduces the similar cases got from OLAP. A case-reduction is defined as a minimal sufficient subset of a set of attributes, which has the same ability to discern concepts as when the full set of attributes is used. A knowledge model in rough set is defined as follows:

$S = <U, C, D, V, f>$

Where

U: a finite set of cases

C: condition attributes set of the cases including problems and solutions

D: decision attributes

V: the value of attribute that a case contains

f: the function that designates the attributes values of each case in U

The case-reduction generation procedure is as follows:

Step1: Analyse the attributes of cases and build knowledge model in rough set for each case
Step2: Establish concise matrix for each case according to logical algorithms Step3: Generate the core of the attributes, i.e. the key attributes that discern cases.

5.2 Knowledge acquisition

Knowledge acquisition is the key to the product knowledge management. Knowledge acquisition includes the capture of the existing knowledge and new knowledge. As mentioned above, product development generates a great deal knowledge resource. The existing knowledge is captured through extracting, aggregating, integrating and digging the database and knowledge bases, which store them. Then they are organized as the subject-oriented multi-dimensional knowledge model (SMM). New knowledge is acquired by machine learning module. The captured new knowledge is organized as SMM. SMM includes six dimensions: what (the object-old cases), where (the revised position of the old case), who (the reviser), when (the revising time), why (the reasons and motivations of revising), and how (the revising methods). SSM is stored together with the old case and the new case for designers to facilitate finding their relations.

5.3 Case example

A tied bill machine has various characteristics, for example it can automatically pack bank notes, can be adjusted the values of bank notes, has warning signals, packing time is less 30s, etc. Consumers may select the desired products based on their requirements (e.g. convenience, safe, price, functions, appearance, etc.). According to consumer requirements, the case retrieval conditions can be expressed as an input model: I= (F, S, P, A, C). Where, F denotes functions of the tied bill machine, fl denotes it can be adjusted the values of bank notes, f2 means it has warning signals; S is automatically packing or not (sl, automatically; s2, not automatically); P is the packing time (pl, packing time<30s; p2, packing time >30s,); A describes the size of the tied bill machine (al, small; a2, middle; a3, large); C represents the price of tied bill machine (cl, expensive; c2, middle; c3, cheap).
Let's assume DR contains six cases in current status and one customer's inquiry is I= (fl, f2, sl, pl, cl, al). OLAP 2D view is used to represent the cases, as shown in Fig.7., in which m1~m6 denote six solutions.

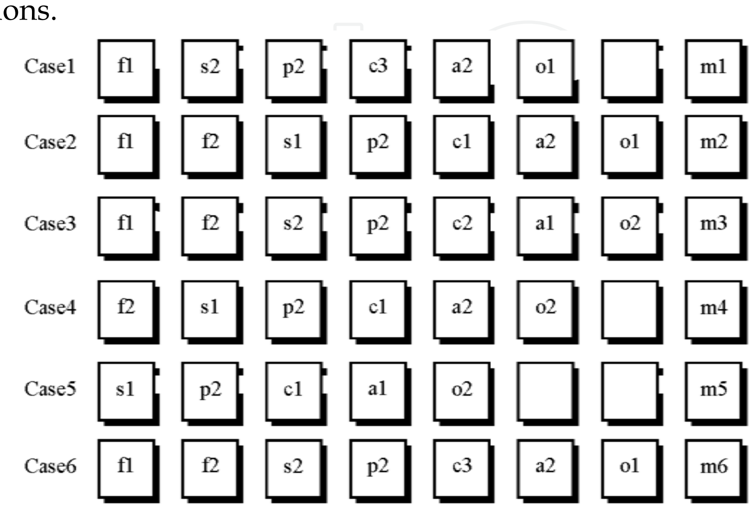


Fig. 7. OLAP 2D view

The inquiry (I) is sent to OLAP which finds all of the six cases contains the attributes that are similar to I. Therefore, the similar set is S (casel, case2, case3, case4, case5, case6). Afterwards, OLAP uses multi-dimensional approach (slice, roll up, drills down, etc.) to find the differences between each case and I in S by going into proper dimension. It discovers only case1 has only one similar attribute, which is thought they have fewer similarities. So case1 is deleted from S. Case6 has only two similar attributes, and it is also deleted from S. Other cases are closely matched between the attributes. So, the answer to the inquiry, the output of OLAP, is S (case2, case3, case4, case5).

It should be noted that the answer couldn't be 100% satisfied, since the packing time (p) is more than 30s. The similar cases are needed to modify. Packing time is directly related to the time of conveying bank notes and felting belts, orientation components, and the correctness of felting positions. In order to find the most similar case in S, a case-reduction method is used to reduce the number and the attributes of the cases in S that will be compared in CBR by NNR. Firstly, a general decision table (Table land Table 2) is incorporated with representing the relationship between condition attributes and decision attributes of the cases. Secondly, the procedure showed in Fig 4 determines the case-reduction and cases, as shown in Table 3. The attributes including felting method, orientation components, felting components, and compaction components are the key attributes that describe the cases. Thirdly, the approach of NNR is used to generate similarity between new cases and the final consistent data in DR.

The most similar case retrieved from NNR is combined with I in case retain, which means the belt component is needed to revise. SMM represents the process knowledge during the variant design of the belt component and the new case will be stored in DR.

Knowledge acquisition is the key to the product knowledge management. Knowledge acquisition includes the capture of the existing knowledge and new knowledge.

attributes		Values of the attributes	
a-structure of belt slot	0- knockdown	a-structure of belt slot	0- knockdown
b-felting method	0-five points	b-felting method	0-five points
c- orientation components	0-driven by electrical power	c- orientation components	0-driven by electrical power
d- felting components	0-driven by double wheels	d- felting components	0-driven by double wheels
e- compaction component	0-double shafts	e- compaction component	0-double shafts
f-width of belt	0-narrow	f-width of belt	0-narrow
g-felting time	0-≤10s	g-felting time	0-≤10s
h-accuracy of orientation	0-not accurate	h-accuracy of orientation	0-not accurate
l-felting location	0- not accurate	l-felting location	0- not accurate
m-feeding time	0-very long	m-feeding time	0-very long

cases	Condition domain						Decision domain			
U	a	b	c	d	e	f	g	h	l	m
1	0	0	0	0	1	1	0	0	0	0
2	0	1	1	1	1	0	1	1	1	1
3	0	1	0	1	0	1	2	1	2	0
4	1	1	1	1	0	1	2	1	2	1
5	0	0	1	1	0	1	0	1	2	0
6	0	1	1	1	0	1	2	2	0	1

case	Condition domain				Decision domain			
U	b	c	d	e	g	h	l	m
1	0	0	0	1	0	0	1	0
2	0	1	1	1	1	1	0	1
3	1	0	1	0	2	1	1	0
4	1	1	1	0	2	1	2	1
5	1	1	1	0	0	1	2	0

Table 1. THE DESCRIPTION OF CONDITION AND DECISION ATTRIBUTES

6. Design Repository

Design Repository (DR) takes the role as a knowledge base where contains all the information and knowledge related to products. Furthermore, it is responsible for collecting, extracting, converting, cleaning, aggregation, and indexing the information and knowledge from various kinds of information and knowledge sources.
The basic rules of knowledge organization are:
(1) constructing proper architectures in order to make knowledge base engine and inference engine quickly find the needed knowledge;
(2) knowledge must be easily maintained.
So the two following data structures are combined to organize design repository.

6.1 Doubly Linked List

Rules are organized by doubly linked list. Fig.8 .shows the structure of doubly linked list [8]. It has two pointers: prior and next. Prior links to the front node in doubly linked list and next links to the next node. The data domain includes three parts: rule number (no), rule presupposition (if) and result (then).

Prior	no	if	then	next

Fig. 8. Doubly linked list

6.2 Orthogonal Linked List

Semantic object network is a directed graph, so graph (a data structure) can be used to organize knowledge. Graph is composed of non-null nodes' set and arcs expressing relations among these nodes. Its definition is as follows [9]:
- G= (V, E)
- V= {vi | vi £ data object}
- E= {(vi, vj) | vi, vj £ V AP (vi, vj)}

G represents a graph; V, the set of nodes in the graph; E, the collection of ordered pairs of vertices, called arcs; P(vi, vj) describes a line connected vertex vi and vertex vj, that is to say an ordered pair (vi, vj) represents an arc. If a pair (vi, vj) £ E composed of any two nodes is ordinal, i.e. lines between vertexes are directional, then the graph is called directed graph.
The information of a graph includes two parts: one is information of vertices in the graph, and the other is information of relations between vertexes or arc information. Therefore graph's storing structure should reflect these two parts' information.
Orthogonal linked list has two linked lists: vertices linked list and arc record linked list. Fig.9. (a) and (b) show the structures of vertices linked list and arc record linked list respectively. Each record in the vertices linked list has a pointer to the arc record linked list that contains the arc information of the node. In arc record linked list, tailvex and headvex represent the locations of arc tail vertex and arc head vertex respectively. Hlink and tlink are two pointers to the next arcs with the same arc head vertex and arc tail vertex respectively. Info points to the arc information. Arcs with the same arc head are in the same linked list; Arcs with the same arc tail are in the same linked list too. Their head nodes are vertices in the vertices linked list. Vertex stores the information on vertices, for example, name of the

Vertex domain	pointer	pointer
Vertex	firstin	firstout

(a) The structure of Vertices Linked List

Arc tail	Head	Information of lines	pointer	pointer
tailvex	headvex	Info	hlink	tlink

(b) The structure of Arc Record Linked List

Fig. 9. The structure of Vertices Linked List and Arc Record Linked List Fig.10. illustrates an orthogonal linked list representation of the graph of PCMSO in Figure 4.

vertex; firstin and firstout are two pointers to the first node with the node as the arc head or the arc tail respectively.

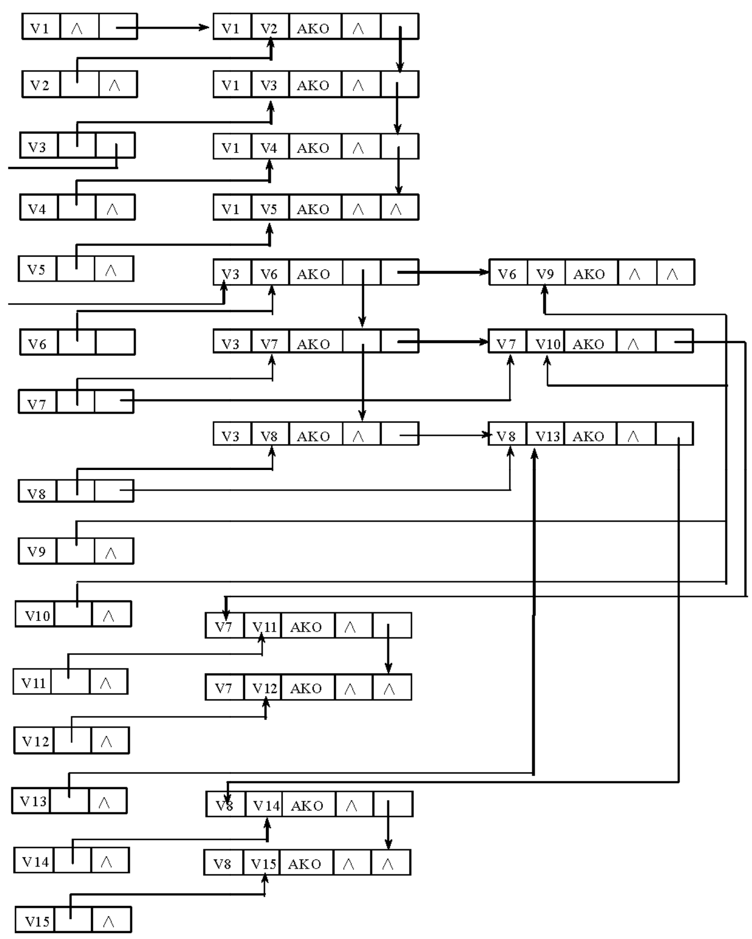


Fig. 10. The Orthogonal Linked List of PCMSOSNs

7. Implementation and application of PKM

According to the above methods, J2EE is utilized to construct a prototype of PKMS supporting the design process of the tied bill machines, in which, web and distributed

database technologies are used to realize the functions of searching, reasoning, browsing, and storing, updating and maintaining knowledge, as is shown in Fig.11.
In order to implement distributed computing technologies, simplify component model based on Internet and improve developing efficiency, J2EE is adopted to establish knowledge base engine, inference engine, updating module and maintaining module. For the sake of simplification, the four modules are built in a Java class package respectively. Servlets are used to exchange data with outer application systems. Searching engine and inference engine use stateless session beans to recall entity beans representing knowledge models. Searching engine and inference engine realize searching and reasoning. Stateless session beans can be shared among clients and cannot be maintained any status messages, so they meet the characteristic of multi-user concurrent access of web-based searching and reasoning. Updating module and user identification module must hold the status messages related to clients, therefore stateful beans are used to recall entity beans, which represent data of knowledge base, and visit the related knowledge in DR through JDBC.
The application tire fulfils the interfaces with product customization platform, CAD/CAM/CAE and etc. Querying requirements of designers are accepted through product customization platform interface. Knowledge on product model and structure is obtained through CAD/CAM/CAE interface. The knowledge is managed and maintained in the application tire.

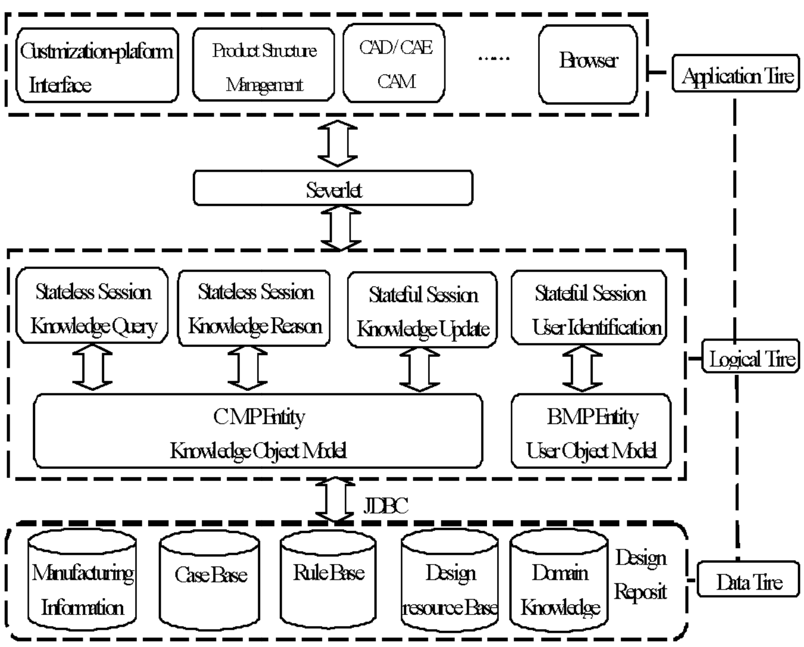


Fig. 11. Framework of Product Knowledge Management System

Fig.12. shows the interface. System has the functions of browsing, searching, updating and etc.
To address those issues described above, we proposed the product knowledge management framework. It provides PACD with an integrated knowledge environment. It uses
The design processes based on PKM are as follows:
(i)Function deployment: implementing function deployment to confirm the conditions of structure deployment according to user's demands;

(ii) Structure deployment: getting the structure tree based on the conditions of structure deployment, using CBR to obtain the similar cases and confirming BOM to finish the design of assembly model;
(iii)Variant design: variant design is necessary if similar cases cannot be acquired. The best variant design can be obtained from PKM;
(iv)New design: new design is needed if variant design cannot meet the needs. Function design, structure design can be implement based on the framework of product knowledge management system.

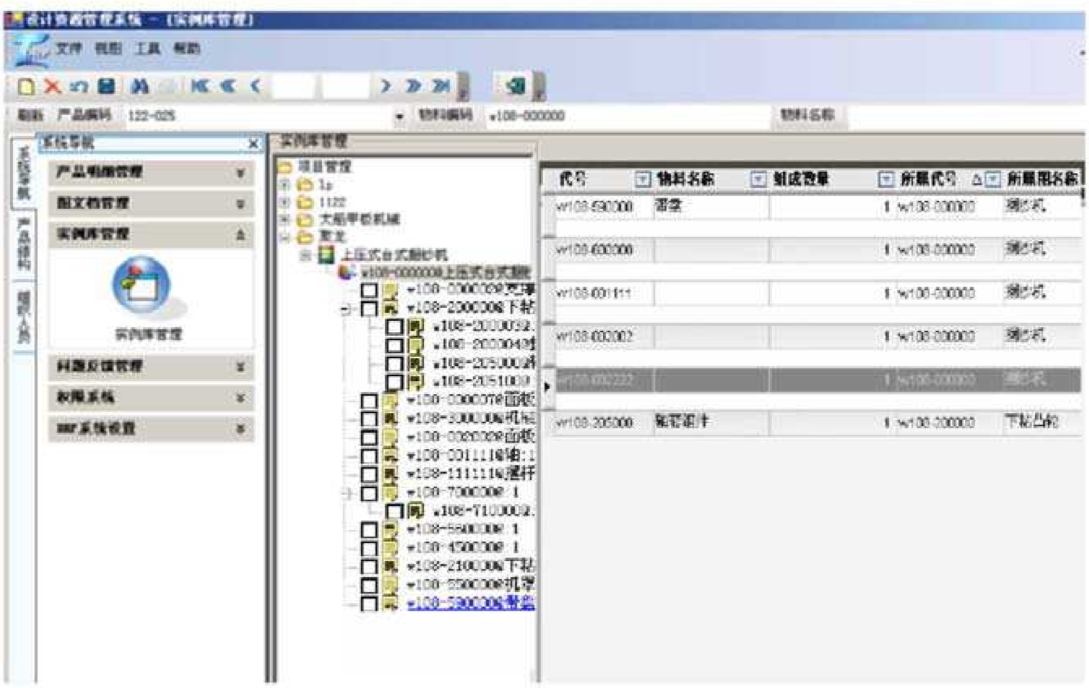


Fig. 12. The Interface of PKMS

8. Conclusions

At present, product knowledge management techniques have not been applied to the real product development process, and the existing document management tools and PDM cannot satisfy the rapid capture and reuse of knowledge during the product agile development processes. In this context, we proposed a product knowledge management framework with DR. A semantic object knowledge model can effectively organize various kinds of knowledge. Knowledge is organized into doubly linked list and orthogonal linked list.Designers could capture and reuse product knowledge by using the OLAP technique, knowledge reduction method, and CBR. Finally machine Learning could acquire new knowledge to make DR alive.

9. References

Cao Jian, Zhang Youliang, Huang Shuangxi, Zhou Xinguang.(1999). Study on Computer Supported Cooperative Design and Its Environment, *Computer Engineering and Applications*, No.5,39-41

Huang C. C, Tseng T L. (2004). Rough set approach to case-based reasoning application, *Expert Systems with Applications,* Vol. 26, 369-385

Huang Wei (2001). The analysis of Semantic Object Model, *Journal of Higher Correspondence Education (Natural Sciences),* Vol.13, No. 5, 40-43

Lai chaoan, Sun yanming (2002). Study on web-based mechanical knowledge acquisition method, *Mechanical design,* 33-36.

Li SiQiang (1999). Semantic Object Modeling Techniques, *Computer Application and Study,* No.5, 40-41

Ni Yihua, Yang Jiangxin, Gu Xinjian,ect on. (2003). Research on CAx Integrated Framework based on Knowledge. *Computer Integrated Manufacturing Systems-CIMS,* 2003, Vol.21, No. 3, 91-94.

Nonaka,I. (1991). The knowledge creating company, *Harvard Business Review.* No. 69, November-December, 96-104

Qi Yuansheng,Peng Hua, Guo Shunsheng, Yang Mingzhong.(2002). Study on the Management of Generalized Design Knowledge and Its Key Technology, *Machine Design,* No.8, 5-9

Simon Szykman. (2000). A web-based system for design artefact modelling. *Design Studies,* Vol.21, No. 2, 145-165

Simon Szykman, Ram D. Sriram .(2000). Design Repositories: Engineering Design's New Knowledge Base. *Intelligent Systems, IEEE,* 47-57.

X.F. Zha. (2001). Knowledge-based approach and system for assembly-oriented design, Part II: the system implementation. *Engineering Applications of Artificial Intelligence,* Vol.14 , 239-254.

Sanja Vranes. (1999). Design knowledge representation in Prolog/Rex. *Engineering Applications of Artificial Intelligence,* Vol.12, No. 7, 221-228.

Soininen T. (2000). An approach to knowledge representation and reasoning for product configuration tasks, *Finnish Academies of Technology,* 20-27.

S R Gorti, A Gupta. (1998). An object-oriented representation for product and design processes. *Computer-Aided Design,* Vol.30, No. 7, 489-501

W.C. Regli1, V.A. Cicirello. (2000). Managing digital libraries for computer-aided design. *Computer-Aided Design,* No. 32 : 119-132.

XB,LIU & Chunli YANG (2003). Method of Building Product Design Knowledge Base, *Computer Integrated Manufacturing Systems-CIMS,* Vol.9, No.8, 204-211

Yang Zijiang, Yang Xiaohu, Li Shanping, Dong Jianxiang.(2001). Research on STEP Feature Model and Its Application in the Integrated Environment of CM, *Journal of Computer Adied Design &Computer Graphics,* Vol.13, No.5, 438-443

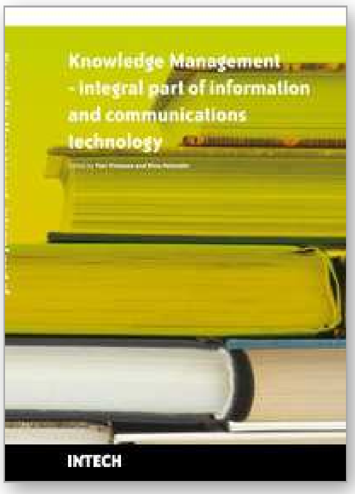
Yu Junhe, Qi Guoning, Wu Zhaotong.(2003). Study on Product Family Structure Model based on Hypergraph, *China Mechanical Engineering,* Vol.14, No.2, 107-110

Zhang Lei, Hou Dianshe. (1998). The Problem Implementing a Conceptual Graph's Tool, *Journal of Xi'an Highway University,* Vol.18, No.2, 107-110

Zhu cheng, Cao wenze. (2002).Knowledge system modelling framework, *Computer engineering,* 3-5

Zhang han, Zhang yongqin.(1999). Deisgn catalog-based computer aided concept design system, Research on computer application, 24-26

Zhang shu, Li aiping (1999). Knowledge supply chain and technology innovation, *China mechanical engineering,* 224-227.



Knowledge Management

Edited by Pasi Virtanen and Nina Helander

ISBN 978-953-7619-94-7
Hard cover, 272 pages
Publisher InTech
Published online 01, March, 2010
Published in print edition March, 2010

This book is a compilation of writings handpicked in esteemed scientific conferences that present the variety of ways to approach this multifaceted phenomenon. In this book, knowledge management is seen as an integral part of information and communications technology (ICT). The topic is first approached from the more general perspective, starting with discussing knowledge management's role as a medium towards increasing productivity in organizations. In the starting chapters of the book, the duality between technology and humans is also taken into account. In the following chapters, one may see the essence and multifaceted nature of knowledge management through branch-specific observations and studies. Towards the end of the book the ontological side of knowledge management is illuminated. The book ends with two special applications of knowledge management.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821