Knowledge Management: System Architectures, Main Functions, and Implementing Techniques

Jun Ma and Matthias Hemmje

Fraunhofer-IPSI, Dolivostrasse 15, Darmstadt D-64293, Germany
{jun.ma,matthias.hemmje}@ipsi.fhg.de

Abstract. Based on the known theoretical models of knowledge management (KM), our investigation on existing KM systems and a prototype system designed for supporting the cooperative research in research institutes, we address three important issues in the design and implementation of KM systems, i.e., the software system architecture of a general KM system, the main functions of the KM systems and the information processing techniques and algorithms that can be used to implement the KM systems.

1. Introduction

Knowledge management is considered to be very important in enhancing the competence of companies and organizations. Larry Prusak (IBM) noted 80 percent of the largest global corporations now have KM projects [14]. Therefore it is an open question whether after the era of management information systems (MIS), KM will become a mainstream technology in the future. Now the progress in the research of semantic web technology has made the documents wrapped in XML or RDF [7,9,27] and ontology languages [22] machine-readable on the Internet. Clearly these techniques have provided KM system developers with a standard interface in knowledge representations and access. Furthermore the progresses in computer science and network technology have made it possible to develop distributed KM systems based on today's Internet/Intranet platforms.

However, some industry observers said KM was a vague concept that would neither deliver what it promised nor add to the bottom line [14]. For example, nowadays document management systems can be declared as KM systems; computer-supported cooperative work (CSCW) systems are also considered to be KM systems. The computer systems consisting of document management systems, knowledge bases, CSCW environments as well as the tools for knowledge discovery are, of course, KM systems. These descriptions on KM systems often make people feel difficult to form a complete concept about KM systems.

In this paper, based on the known theoretical models of KM, our investigation on existing KM systems and a prototype system designed for supporting the co-research and software development in research institutes, we address three important issues in the design and implementation of KM systems, i.e., the software system architecture of a general KM system, the main functions of the subsystems of the KM system and the information processing techniques and algorithms that can be used to implement

Y. Han, S. Tai, and D. Wikarski (Eds.): EDCIS 2002, LNCS 2480, pp. 155-167, 2002.

the KM system. We propose a software system architecture designed as service-oriented, and demonstrate the methods and algorithms that can be used to implement the general KM systems as well.

2. KM Theory

In KM theory, data are observations or facts; information is the data organized and categorized into patterns to create meaning; and knowledge is information put to productive use, enabling correct action and helping people to produce innovation. Knowledge also refers to what we have been acquainted, e.g., working experience, problem-solving methods, customers' profiles etc. Usually people classify knowledge into two categories, i.e., *explicit* and *tacit* knowledge, where *explicit* knowledge is knowledge that has been articulated in the form of text, tables, diagrams and so on, while *tacit* knowledge is knowledge that cannot be articulated, which mainly resides in people' brains. However there is no a clear boundary between the two sets for any application domain. In contrast, there is a nonempty intersection set of the two sets. The knowledge in the intersection set is called *implicit* knowledge, i.e., the knowledge that can possibly be articulated but has not been done. A KM project is to establish an open culture and improve ways of cooperative work in an organization. A concrete goal is to establish a computer system called KM system, which will be used to set up a framework and tool set for improving the organization's knowledge infrastructure.

The most popular theoretical models on KM have been summarized in Fig. 1 [26]. These models provide clear descriptions of knowledge-life-cycles.

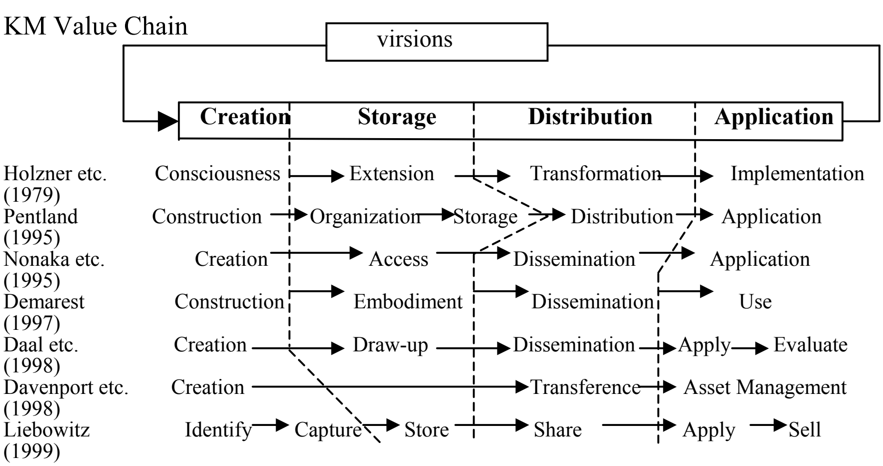

Fig. 1. KM models and KM-Value-Chain.

However these models cannot be used to develop a KM system directly because system developers need to know what the software system architecture of a KM system is, the basic functions of the KM system and the information processing technologies that can be used to implement the KM system.

3. Related Work

The software system architectures of KM systems and the implementation techniques in the design and implementation of KM systems have been studied before. For example, KM systems are classified as *work-flow* (*processing*) oriented [11,13], where knowledge creation, storage, distribution and reuse are implemented based on the basic work-flows of enterprises, *problem-solving* oriented [7], where knowledge creation, distribution and reuse are carried out along with the process of disintegrating the goal of problem-solving into sub-goals, *knowledge-creation* oriented [3,6, 21,24], where the main task of KM systems focus on enhancing organizational learning and discovering new knowledge. We think the three types of system architectures often dependent on the working flows and the administration of concrete enterprises or organizations. Therefore we consider service-oriented system architecture seems more suitable to be a general one in the design of general KM systems. We motivate this below.

We investigated many successful and failing KM systems, we found that KM applications are not limited to the innovative organizations only, such as software companies and research institutes, where knowledge/information is considered to be the primary basis for business development and success. KM techniques have also been applied to the knowledge management of many public sectors and service organizations[1,3,4,7,9,11,12,13,14,20,21,23,25]. In most KM systems knowledge usually does not refer to scientific or technical innovation directly, but the management of staff members, cooperation, business transactions and learning and training. Concrertely speaking the knowledge in these systems is often about the customer profiles, synergy relationships, administration and business experience and all others that can help staff members to act more successfully. The usage of this kind of knowledge is to enhance the working efficiency, decrease expense, avoids faults and reinventing, reuse knowledge to new problems as well as provide various services in the process of people's innovation activities. An investigation on KM systems within the experts involved in the Social Security Administration's benefit rate increase process [16] is shown in Tab. 1. We think it is of the representative among the known KM systems.

Table 1.

Major benefits of KM	percentage of responses
Increased innovation	20%
Practice and process improvement	60%
Increased customer satisfaction	30%
Enhanced employee capability and organizational learning	50%
Improved efficiencies in writing reports and responding to inquiries	10%
Lower learning gap	10%

4. A General System Architecture for the Design of KM Systems

Based on the discussion above and the classification of knowledge mentioned in section 2, we consider a complete KM system should consist of at least three subsystems, i.e. a subsystem named E-KM, which manages the explicit knowledge and provides basic knowledge management functions; a subsystem named T-KM, which manages the tacit knowledge in terms of providing the cooperative working environment and intelligent services for staff members and partners to work together; and finally an alternative subsystem named I-KM, which provides the tools and software applications for knowledge discovery. The E-KM forms the knowledge infrastructure of an organization, while a T-KM provides intelligent services based on the information and knowledge stored in the corresponding E-KM. I-KM is used to find new knowledge by mining the web pages on the Internet/Intranet as well as the data in the databases and knowledge bases of the E-KM and T-KM. However since the applications of today's knowledge discovery techniques are still limited, we believe it is better to consider I-KM to be an alternative subsystem consisting of the separate techniques in the design of KM systems.

Clearly for any a KM system S, $F(S) \subseteq F(E\text{-}KM) \cup F(T\text{-}KM) \cup F(I\text{-}KM)$, where F(A) represents the functions provided by a computer application system A.

Let us discuss the basic modules and the system architectures of E-KM and T-KM in detail. However, we think it is difficult to do the similar discussion on I-KM because we feel that the techniques used in knowledge discovery lack a tight connection for the time being.

4.1 A Design of an E-KM Subsystem

An E-KM subsystem mainly consists of document management systems, case bases, knowledge bases and intelligent services. The main functions are listed below.

- Information/Knowledge sharing and the intelligent services for information/ knowledge reuse.
- Knowledge audit: which looks at a targeted area and identifies which knowledge is needed and available for that area, which knowledge is missing, who has the knowledge.
- Knowledge learning support: which helps users to find learning documents and arrange study corresponding to their learning purposes and education backgrounds.
- Searching engines: The searching engine helps users to find former working experience matching the searching requirement of users.
- Knowledge maps: graphic display of knowledge flows and knowledge synergy among partners. The nodes in the graph denote persons, the research groups, divisions and partners worldwide; the arcs represent the knowledge flows or synergy relationships. Different colors describe different kinds of synergies or knowledge flows.
- Yellow pages: It stores the communication information of members and partners, e.g., the name, position, research interests, projects joined, phone, fax and email, in detail.

- Knowledge filters: A knowledge filter is used to distinguish valuable and less valuable knowledge. The judgment is made based on the perspectives of experts who are aware of it and the evaluating factors that people want to emphasize, e.g., "creativity", "reliability", "importance" and so forth.

Based on our prototype system, we propose a software system architecture of E-KM subsystem in Fig. 2. There are four submenus, where MIS offers the basic services in information/knowledge storage, edition, retrieval and indexing, statistical calculation and keeps the profiles of members and groups; the submenu of *searching pilot* helps users to find knowledge stored in the data/knowledge bases, where documents in the system are organized in document ontology [8,9,22], i.e., represented in taxonomic graphs without directed cycles. The documents in the system are classified into several categories based on application domains, which are denoted by *D1, D2,…,Dn*, e.g., *e-learning, multimedia applications, user interface design, e-business* and *KM systems*. Projects, which mean the names of projects here, are linked to the domains they belong to, and the documents belonging to a project are classified further based on document types, e.g., *reports, working plans, while papers, formal publications, group profiles, meeting minutes, system demonstrations, handbooks* as well as the *frequently asked questions (FAQ).* Subclass-of relations are used to set the semantic relationships of the concepts of categories and real documents in the system.

The documents about *best practices, lessons* and the *projects* carried out before are represented by meta-data in RDF or XML, where the meta-data annotate the semantic relationships among the entities in a document, where an entity may be a demo, a literal report, and a piece of graphic and image. The real documents corresponding to the entity descriptions are dispersed across a number of servers.

4.2 A Design of A T-KM

The management of tacit knowledge refers to many disciplines, e.g., management and computer science as well as psychology. We only discuss this issue from the viewpoint of software system developers. In principle a CSCW environment helps staff members and the partners in the community to exchange their know-how and

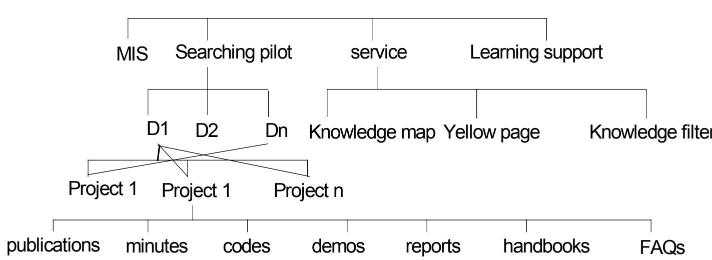

Fig. 2. A system architecture of the E-KM in our prototype system.

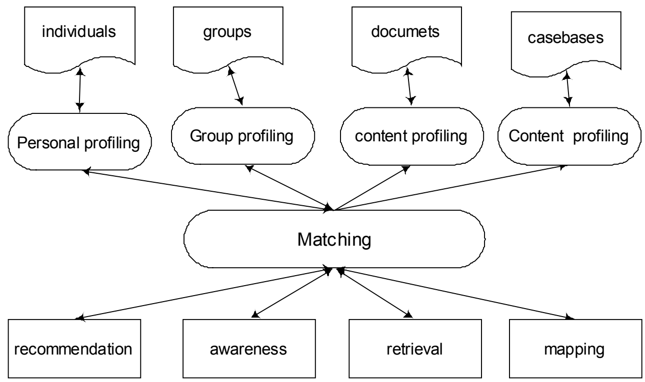

Fig. 3. A system architecture of a T-KM.

knowledge in various ways, e.g., emails, videoconference systems etc. It can be implemented based on a commercial or developed CSCW platform, e.g. Lotus Note. However the services supporting KM, e.g., the functions of awareness and recommendations for finding experts and expertise, usually have to be developed independently. A system architecture of a T-KM is shown in Fig. 3.

The services of the T-KM are built based on *matching* that connects the profiling of individuals and groups, as well as the profiling of the descriptions of documents and case bases through the descriptions of domains and projects information. We summarize the some concrete matching methods below. These matching methods were used in the design of cooperative filters [5]. Now we use them with a little change in the design of T-KM systems.

1. If the profiles of individuals, groups, documents, domains and cases are represented by keyword or phrase sets that describe research interests in special domains, the similarity of two profiles is calculated by formula 1.

$$\text{Similarity}(A,B) = |A \cap B| / |A \cup B| \quad (1)$$

where A and B represent two keyword sets.

2. If the profiles are represented by frames f_1 and f_2, where a frame describes the basic personal information, research interests, working experience and education backgrounds, then the similarity of two profiles is calculated by formula 2.

$$\text{Similarity}(f_1,f_2) = \frac{\sum \textit{the weight of the matched slots}}{\sum \textit{the weight of all slots}} \quad (2)$$

3. If a profile is represents by an $m\times n$ matrix R, where R is a relation matrix of members×interested domains (projects × domains, documents × domains and document types × domains, etc.), such that $r_{i,j} = 1$ if the ith member has the interests in the jth domain. Then the similarity of two members in research interests is calculated by formula 3

$$\text{Similarity}(r_i, r_j) = \cos(r_i, r_j) = \frac{r_i \circ r_j}{\|r_i\| \times \|r_j\|} \quad (3)$$

where "$\circ$" denotes the dot-product of the two raw vectors of R, or

$$\text{Similarity}(r_i, r_j) = \text{corr}(r_i, r_j) = \frac{\sum_k (r_{ik} - \overline{r_i})(r_{jk} - \overline{r_j})}{\sqrt{\sum_k (r_{ik} - \overline{r_i})^2 \sum_k (r_{jk} - \overline{r_j})^2}} \quad (4)$$

Formula 3 is called Cosine and formula 4 is called Correlation. Since the computation of Cosine is much simpler than that of Correlation, the computation of cosine is adopted more in practice. However the efficiency of above formulae depends on concrete problems, applications and data.

The main functions of a T-KM subsystem are listed below.

- Recommendation. (1) Recommends experts in a special domain; (2) Recommends team members for a new project; (3) Recommends documents that represent the best practices and lesions to the system developers who are undertaking a similar new project.
- Awareness. (1) Sharing a common calendar. In order to arrange meetings, discussion and all other common activities, a common calendar recording the activities of staff members in a division is set up. The calendar also records the common critical dates, e.g., the deadlines of conference and the date for the visitors outside. (2) Document sharing. The members in a (virtual) team share a file space, where the members can upload documents, drawing, images, etc to the file space and the system can inform others to download the documents in time.
- Retrieval. A searching engine for scientific publications through the Internet is available, which can help staff members to decrease the irrelative URLs and increase the search speed by utilizing several famous search engines, e.g. Google, Yahoo, etc. to do the information retrieval in parallel [10].
- Mapping. Displaying various relationships graphically as well as depicting the roles of different people in the graph. These relationships include the synergy relationships among individuals and groups.

Although Fig. 3 provides a high level description of T-KM, it shows the relationship between E-KM and T-KM. Clearly, E-KM is the fundamental subsystem of a KM system because the design and implementation of T-KM is based on document management of E-KM.

5. Our KM Practice

Our KM practice mainly refers to the project called Delite-Online, which is a virtual information and knowledge management systems for the division of Delite of Fraunhofer IPSI [17]. Fig. 4 shows the portal of the system and Fig. 5 is the GUI about the calendars and internal schedules of divisions.

The software system architecture and main functions provided by the system have been discussed in sections 3.1 and 3.2. For the time being, most of the functions have been implemented based on the similar methods and technologies that have been described in [3,4,28]. Therefore, we only introduce an algorithm for knowledge filter and a mechanism supporting learning and training in enterprises, which seem different from the known ones essentially.

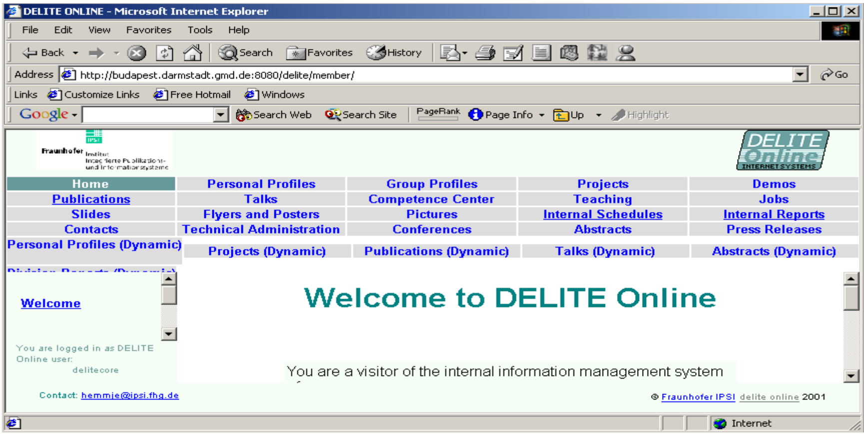

Fig. 4. The portal of Delite-Online.

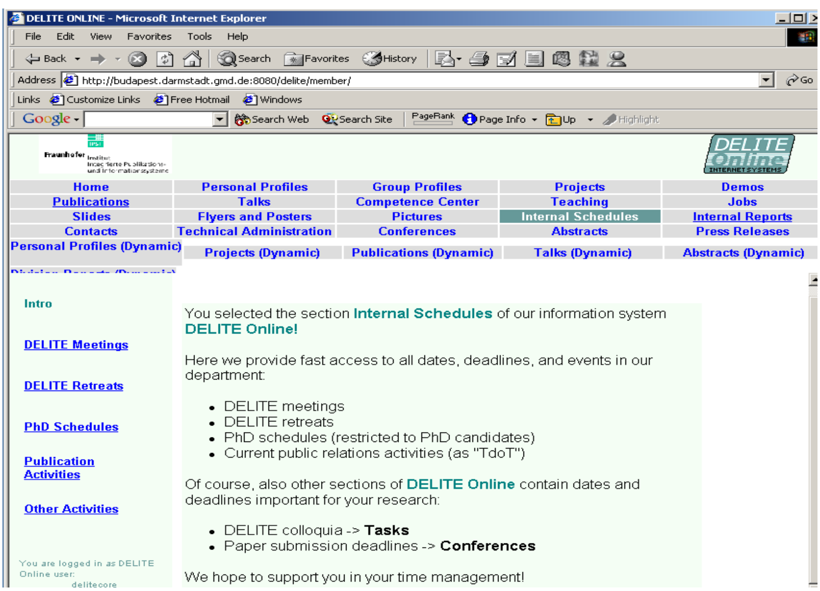

Fig. 5. Calendars and internal schedules of divisions.

5.1 A Novel Knowledge Filter

Knowledge filter is used in (1) ranking files based on the comments of experts or peers; (2) recommender expertise and files. Although many filters or recommender systems have been studied for e-commerce [5], it is still hard to use them directly for document management because only one evaluation factor is considered in these algorithms. In document management multi-factors have to be considered in document evaluation, say *originality, creativity, presentation, relativity* and so on. Furthermore, often some factors are more important than others in the document retrieval for a special purpose, e.g., the factor of *originality* is more important than *presentation*. We propose an algorithm that considers both multi-factors and the weights of the factor. The formal description is given below.

Let P represent a file, e.g., which is a proposal or business strategy. We design an m×n matrix to collect the comments for P from experts. Let U = $\{u_1, u_2, ..., u_n\}$, where u_i, 1≤ i ≤ n, is an evaluation factor, such as, *originality, creativity, presentation, relativity* and so on. The comment set V = $\{v_1, v_2, ..., v_m\}$, where v_i is a comment, 1≤ i ≤ m, e.g. "*bad", "normal", "good" and "excellent"*. If an expert chooses a comment v_i to evaluation factor u_j for P, then he/she gives a mark at the position (*i,j*) of the table. Following algorithm gives a synthetic comment by fuzzy and statistic computation to determine whether the knowledge is good enough to be added to knowledge bases. The detail will be discussed below.

Algorithm 1 (synthetic comment computation)

Input:

1. A file P;
2. The evaluation factor set U=$\{u_1, u_2, ..., u_n\}$, where u_i, 1≤ i ≤ n, is an evaluation factor;
3. The comment set V =$\{v_1, v_2, ..., v_m\}$, where v_i is a comment, 1≤ i ≤ m.
4. A fuzzy vector X= $(x_1/u_1\ ,\ x_2/u_2\ ,\ \cdots\ ,\ x_n/u_n)$, such that, $\sum_1^n x_i = 1$. $x_i \geq 0$, *i =1,2,…,n*. x_i represents the importance of the factor u_i in the evaluation.
5. The experts' comments on P.

Output: A synthetic comment of P.

Step 1. After collecting the comments from all referees for a file P, calculate following fuzzy sets:

$(\mu_{i1}/v_1\ ,\ \mu_{i2}/v_2\ ,\ \ldots\ ,\ \mu_{im}/v_m)$, 0≤ μ_{ij} ≤1, *i* =1,2,…*n;* *j*=1,2,…*m*; Where μ_{ij} = (the number of persons who give v_j to u_i)/ (total number of referees)× 100%. *N* fuzzy sets are represented in a matrix $E = (\mu_{ij})_{n\times m}$.

Step 2. Calculate following vector

$(y_1/v_1\ ,\ y_2/v_2\ ,\ \ldots\ ,\ y_m/v_m)$ by following formula.

$$y_i = \bigvee_{i=1}^{n} (x_i \wedge \mu_{ij})\ ,\ j = 1\ ,\ 2\ ,\ ...,\ m.$$

Where ∨ and ∧ represent a suitable operation, e.g. a∨b = max(*a,b*); and $a \wedge b$ = min(*a,b*);

Step 3. If y_k is of the maximum in set { $y_1, y_2, ...y_m$ }, then output v_k as the synthetic comment.

P will be added to knowledge base if v_k is good enough. The principle of the algorithm can be stated as follows. We do the statistics computation in the step 2, where the percentage of the comment v_j gotten in all given comments on evaluation factor u_i is calculated, $1 \le i \le n$ and $1 \le j \le m$. In step 3, based on the principle of fuzzy processing technique, a fuzzy vector X = $(x_1/u_1\ ,\ x_2/u_2\ ,\ ...,\ x_n/u_n)$ is used, where x_i represents the importance of evaluation factor u_i in the computation of a synthetic comment on a proposal P. x_i is as a weight value in the calculation of the mathematics expectation of each v_j, $1 \le j \le m$ also. Clearly the computation of mathematics expectation of v_i, $1 \le i \le m$, considers both the comments from all referees and the importance of each evaluation factor. We choose the one with the maximum mathematics expectation among all comments as the final comment on P.

For our knowledge, this algorithm is the most complicate computation among the design of the filters given in [5,19]. For the time being, the algorithm is only used to rank files based on the comments of experts or peers and recommender expertise. In the future we want to recommend documents according to the ranking and measure the accuracy of the ranking in practice.

5.2 A Mechanism for Supporting Learning & Training

The system architecture of the mechanism for supporting e-learning systems in our system design is shown in Fig. 6, where the module *scheduling* produces a learning material list called *Curriculum Schedule* (CS) to a user based on his or her learning purposes gotten by human-computer interaction through the module *interface.* The education knowledge represented by Precedence Relation Graphs (PRGs) and the descriptions on the learning materials (textbooks, while papers, etc.) stored in *Databases*. A CS tells the user what materials he or she should study based on his or

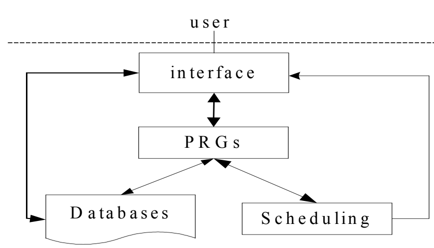

Fig. 6. An architecture of the mechanism supporting e-Learning.

her education background and learning purpose, and suggest learning the courses in the order listed in the CS.

A term T_i consists of following data, which consist with the metadata of T_i stored in Databases.

1. Name of the learning material
2. Author Name
3. Type identifier (book, paper, video,…, manual)
4. Publisher
5. Published time (mm/dd/yy)
6. Level identifier
7. Internal number
8. Application domain
9. URL (for accessing the syllabus of the document)

The *precedence relation* "<" on these documents is defined based on the education and training knowledge, e.g., the relationships among the contents of the materials and the empirical recommendations from experts and lecturers. A PRG is a directed graph G(V,A) consists of a vertex set V and an arc set A. A vertex is triple (No, Name, Level), where No denotes a vertex of V, Name = Document Name in databases; level, $1 \leq$ level ≤ 3, represents the depth of the content of the documents, i.e. 1 = *elementary level*; 2 = *meddle* level and 3 = *professional* level. An arc $<u,v>$ $\in$ A iff $u < v$.

Definition 1. An operation called Topology Sort on a directed graph G(V,A) without directed cycles is to generate a permutation of the vertices of G, such that, if vertex x is ahead of y in the output iff arc $<y, x>$ is not in A.

An O(|V|+|A|) topological sort algorithm was prompted in [2]. Let TS(G,L) denote the a topology sort on a G(V,A) and L represent the output, and further assume the profiles of users, e.g. the courses studied before, can be accessed. Algorithm 2 describes the main steps to generate a CS. The correctness of it is obvious and it is easy to prove that the time complexity of it is O(|V|+|A|) as well.

Algorithm 2 (Scheduling)

Output: A curriculum schedule

1. Getting an ID, an integer LEVEL and the purpose of learning/training of a user, then determine a PRG $G_i \in$ PRGs meeting the learning requirements of the user.
2. Delete the courses $\in G_i \cap$ Course(ID) as well as the connected arcs from G_i, where Course(ID) denotes the courses that the user has studied.
3. Delete all textbooks whose levels $\neq$ LEVEL from G_i further;
4. Call TS(G_i,L) to generated a topology sort on G_i, where table L consists of the vertices of G_i.
5. Output a CS based on L, where each item of the CS consists of nine metadata that we just discussed above.

The algorithm will output a solution based on the acquired searching requirement unless system cannot find corresponding PRG in the knowledge bases of our system. However, since PRGs are established based on general education and empirical knowledge, the output of the algorithm cannot ensure it is the optimal arrangement for learning for each system user.

6. Conclusions and Future Work

In this paper we address three important issues in the design and implementation of KM systems, i.e., the software system architecture, main functions in the subsystems of KM systems as well as the techniques to implement the KM systems based on the software system architecture. Although the examples given in the discussion come from our prototype system designed for supporting the KM in research institutes in the application domain of software system development, the software system architecture and implementing techniques are, in fact, independent of our research institute. The system architectures of KM systems for other applications can be deduced from the given software architecture, say changing the organization of documents in order to suit the situation of organizations. The software architecture of general KM systems is proposed based on our investigation on known successful KM systems and our KM practice. We argued that general software system architectures for KM systems should be service-oriented and human-centered. Based on this viewpoint, a general KM system consists of three subsystems, i.e., the subsystem named E-KM, which manages the explicit knowledge and provides basic knowledge management functions, the subsystem named T-KM, which manages the tacit knowledge in terms of providing the cooperative working environment and intelligent services for staff members and partners to work together, and the subsystem named I-KM, which provides the tools for knowledge. The software system architecture shows that E-KM is the knowledge infrastructure of an organization, T-KM is established based on E-KM. I-KM provide the basic services for supporting human innovational activities based on both E-KM and T-KM, however, it is a relative independent subsystem. We showed our knowledge filter and a mechanism for supporting self-learning in the implementation of our prototype in this paper. We think that the knowledge filter can also be used for e-commerce and the mechanism for supporting learning can be utilized in enterprise learning and training independently.

Our future work is to complete the implementation of some modules belonging to T-KM and I-KM in our prototype system further, and try to add more functions belonging to I-KM into the prototype system.

References

[1] Abecker A., Bernardi A., Hinkelmann K., Kehn O. and Sintek M., Toward a technology for organizational Memory, IEEE Intelligent systems, May/June 1998, 40-48.

[2] A.V.Aho, J.E.Hopcrof and J.D.Ullman, The Design and Analysis of Computer Algorithms, Addison-Wesley Publishing Company, Inc, 1974, pp 70.

[3] Borghoff U.M and Pareschi R., Information Technology for Knowledge Management, Springer-Verlag Press, 1998, 1-8.

[4] Cladwell N.H.M, Clarkson P.J., Rodgers P.A. and Huxor A.P., Web-based Knowledge Management for Distributed Design, IEEE Intelligent Systems, May/June 2000, 40-47.

[5] Communications of ACM, march 1997, 40(3).

[6] Fayyad U.M, etc. edited, Advances in Knowledge Discovery and Data Mining, AAAI Press, California USA, 1996.

[7] Gresse C von Wangenheim, Althoff K.D. and Barcia R.M., Goal-Oriented and Similarity-Based Retrieval of Software Engineering Experienceware, Proceedings of 11th International Conference on Software Engineering and Knowledge Engineering, Eds by G.Ruhe and F.Bomarius, Kaiserslautern, Germany, June 1999, Lecture Notes in Computer Science 1756, pp 118-141.
[8] Guarino N., Understanding, Building, using ontologies. Int. J. Hum. Comput. Stud. 46(2/3):293-310,1997.
[9] Guerrero L.A. and Fuller D.A., A Web-based OO platform for the development of multimedia collaborative applications, Decision Support Systems, 27, 1999, 255-268.
[10] L.Huang, M.Hemmje, E.J.Neuhold, ADMIRE: An Adaptive Data Model for Meta Search Engines, Computer Networks, 33(1-6), 2000, 431-448.
[11].Kappel G., Schott S.R., and Retschitzegger W., Proceedings on Coordination in Workflow Management Systems A Rule-Based Approach, Coordination Technology for Collaborative Applications,1997, Eds. Wolfram Conen, G.Neumann ,Singapore, Springer-Verlag press, 100-119.
[12] Khalifa M. and Kwok R.C.W., Remote learning technologies: effectiveness of hypertext and GSS, Decision Support Systems, 26, 1995, 195-207.
[13] Kim K.H. and Paik S.K. Practical Experiences and Requirements on Workflow, Proceedings on Coordination Technology for Collaborative Applications, Wolfram Conen and G.Neumann (Eds) ,Singapore, Springer-Verlag press, 1997, pp 145-160.
[14] G.Lawton , Knowledge Management: Ready for Prime Time? IEEE Computer, 34(2), 2001, 12-14.
[15] Leary D.E.O, Using AI in knowledge management: Knowledge Bases and Ontologies, IEEE Intelligent systems, May/June 1998, 34-39.
[16] Liebowitz J. Knowledge Management-Learning from knowledge engineering, CRC Pess LLC, 2001, 93-102.
[17] Ma J. and Hemmje M., Developing Knowledge Management Systems Step by Step, The Proceedings of the 2nd European Conference on Knowledge Management, Edit, Dan Remenyi, Bled, Slovenia, 2001, 301-310.
[18].Mandviwalla M and Khan S., Collaborative Object Workspaces(COWS): exploring the integration of collaboration technology, Decision Support Systems, 27, 1999, 241-254.
[19] McDonald, D. W. and M.S. Ackerman. Expertise Recommender: A Flexible Recommendation Architecture. Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW '00), 2000: 231-240.
[20] Mertins K. H, Heisig S.P, and Vorveck J., Knowledge management-Best practices in Europe, Springer-Verlag, Berlin, Germany, 2001.
[21] I.Nonaka and H.Takeuchi, The knowledge Creating Company: How Japanese Companies Create the Dynamics of Innovation, New York: Oxford Univ. Press, 1995.
[22] Rabarijaona A., Dieng R., Corby O. and Inria R.O., Building and Searching an XML-Based Corporate Memory, IEEE Intelligent systems, May/June 2000, 56-63.
[23] E.Motta, Reusable Components for Knowledge Modeling. IOS, Amsterdam, The Netherlands, 1999.
[24] C.Rouveirol, A Knowledge-Level Model of a Configurable Learning System, IEEE Expert, August 1996, 50-58.
[25] P.M.Schiefloe and T. G Syvertsen, Proceedings on Coordination in Knowledge-Intensive Organizations, Coordination Technology for Collaborative Applications, 1997, Eds. Wolfram Conen, G.Neumann ,Singapore, Springer-Verlag press, 9-23.
[26] Shin M.,.Holden T, and Schmidt R.A. From knowledge theory to management practice: towards an integrated approach, Information Processing & Management, 37, (2001), 335-355.
[27] XML and RDF, www.w3.org/TR/REC.
[28] Yang C. Steifield C. and Pfaff B, Virtual team awareness and groupware support: an evaluation of the TeamSCOPE system, Int. J. Human-Computer Studies, 56, 2002, 109-126.