Implementation of Task-based Knowledge Support System

Student: Kuen-Shieh Yang **Advisor:** Dr. Duen-Ren Liu

Institute of Information Management

National Chiao Tung University

Abstract

Managing Knowledge within and across organizations is considered as a prominent activity for creating sustainable competitive advantages in today's business environments. The operations and management activities of enterprises are mainly based on tasks, in which organizational workers perform various tasks to achieve business goals. In task-based business environments, an important issue of deploying KMS is how to support task-relevant knowledge by considering the characteristics of tasks in organizations. This work implements a task-based platform to support knowledge management activities, including knowledge creation, management and dissemination. The objective of this system is to effectively manage task-relevant explicit (codified) knowledge and tacit (human resource) knowledge. A flexible task-based knowledge support system is developed using object-oriented techniques.

Keywords：Knowledge management, task-based knowledge support, profile, knowledge retrieval, knowledge sharing

實作以工作為基礎之知識支援系統

研究生：楊昆學　　　　　　　　　　　　指導教授: 劉敦仁 博士

國立交通大學資訊管理所

摘要

在今日企業環境中，管理跨組織或組織中之知識被視為創造持續之競爭利益的重要活動。企業之操作與管理行為主要以工作為基礎，組織工作者執行不同工作以達成企業目標。在以工作為基礎的環境中，如何藉由考慮到組織工作特性以支援相關工作知識為部署知識管理系統的一個重要議題。本研究實作一個以工作為基礎之平台以支援如知識創造、管理與散播之知識管理活動。本系統以有效的管理相關工作之外顯（經編撰）知識與內隱（人力資源）知識為目標。本研究使用物件導向的觀念，實作一個彈性並且以工作為基礎的知識支援系統。

關鍵字：知識管理、以工作為基礎之知識支援、特徵檔案、知識搜尋、知識分享

Table of Content

List of Figures

1. Introduction

1.1 Research Background

Knowledge management systems (KMS) are emerging as important and practical tools for enterprises to improve decision making effectively. Knowledge creation, management, and sharing are three keys and are associated across various organizations for managing knowledge (Davenport and Prusak 1998; Nonaka, 1994). Knowledge management systems (KMS) supports three activities and are manifested in a variety of implementations such as formal training, knowledge repositories, knowledge fairs, communities of practices and talk rooms (Davenport, De Long & Beers 1997).

The information technology enables the activity of knowledge management to be developed based on the tools supporting the Data Warehouse, OLAP, Data Mining, Information Retrieval, Decision Support System and so on. Adopting these database technologies can support different types of knowledge management systems, and the Gartner Group is a well known multi-tier knowledge management systems (Gartner Group 1999) which noted that knowledge retrieval is the newest addition and core components on the existing IT. Notably, two dimensions are considered in this function, which are "*semantic*" dimension and "*collaboration*" dimension. The semantic dimension concentrates on how to structure knowledge and provide a knowledge guide to people that is similar to the concept of "Yellow Pages." The collaboration dimension is to provide "value recommendation" which comes out from the people (e.g., domain experts, key workers and decision makers) in a collaborative working environment. A collaborative working environment can be achieved by constructing practice of community, which is widely discussed in Recommender System adopted in Electronic Commerce (Schafer, Konstan & Riedl 2000). In addition, the concepts of exploration and exploitation for organization learning

(March 1991) shades additional light on the KMS and are considered as two important parameters to capture the related dimension of "breadth" or "new possibility" and "wideness" or "existing resources" (Benner & Tushman 2000).

Traditional information system assumes people performing repeated and routine works. Therefore, they usually transfer knowledge through document management system or searchable database. This assumption limits the ability of workers to learn and innovate and inhibit organization to increase its core competence. Ellis (1991) defined *"Computer-based systems that support groups of people engaged in a common task (or goal) and that provide an interface to a shared environment*." Computer-Supported Cooperative Work (CSCW) emphasize on the power of computer system to help group of people perform the tasks (Rodden, 1991).

1.2 Research Motivation and Objective

(1) Research Motivation

Our research is mainly to achieve the organizational knowledge reuse through the knowledge retrieval and knowledge sharing. We evaluate the characteristics of information needs of knowledge workers. In addition, we analyze the characteristics of task-based working environments. Accordingly, a task-based knowledge support platform is set up to achieve the objective of knowledge management activities, creating, organization and sharing. In addition to promote the practice of management activities, it is also a more effective and flexible way to achieve the goal of knowledge management.

Accordingly, the motivation of this work is listed as follows:

- Fully and economic reusing valuable knowledge assets by deploying Knowledge Management System.
- Supporting task-relevant knowledge by considering the characteristics of tasks in organizations.
- Enhancing knowledge retrieval and promoting knowledge sharing to support knowledge workers performing knowledge-intensive task in workplaces.

(2) Research Objective

Our research is mainly to establish a knowledge support system. A real domain, a research institute is the target of this work. The system is built upon a web-based environment, and coded in an object-oriented style to capture the features in the real world. Since documents are the important explicit knowledge resources in an organization, we adopt the standard Information Retrieval (IR) technology to accomplish the text processing. Besides, Database Management (DBMS), Information Filtering (IF) and Information Extraction (IE) are also important techniques to support the work.

Accordingly, the objective of this work is listed as follows:

- The overall design of system architecture departs from how to provide relevant and necessary task-relevant knowledge to support workers accomplish tasks. Therefore, the proposed system mainly focuses on managing task-relevant knowledge items and it could be a more effective knowledge management system in task-intensive business environments.
- The task profiling mechanism is deployed in the system to provide task-based intelligent services. The task profile modeling considers task characteristics and domain information structure in task-based business environment to leverage system's knowledge retrieval capability.
- The knowledge support portal (K-Support portal) is set up to support collaborative work in task-based environments. It also facilities the task-based peer groups to share their knowledge to promote the knowledge dissemination.

1.3 Content Organization

The rest of this paper is organized as follows. Section 2 surveys the related work, including Knowledge Management, Task-based Knowledge Management, Information Retrieval and Information Sharing. Rational to design task-based knowledge support system and the overall system architecture are illustrated in Section 3. The method to achieve knowledge support is presented in Section 4. The detail of system implementation is described in Section 5. Next, Section 6 presents the system demonstration with associated scenarios and interfaces. The conclusion and future research work will be concluded in Section 7.

2. Related Works

2.1 Knowledge Management and Knowledge Management System

Managing Knowledge within and across organizations is considered as a prominent activity for creating sustainable advantages in today's competitive business environments. Knowledge Management (KM) is a cycle, sometimes repeated process, which generally includes creation, management and sharing activities. (Davenport & Prusak, 1998; Nonaka, 1994; Wiig, 1993). Information technology (IT) supports different types of knowledge management systems by adopting tools, including Data Warehouse, OLAP, Data Mining, Document Management System, Expert Systems and so on (Hahn & Subramani, 2000). Generally, information technologies (ITs) mainly focus on two dimensions, explicit and tacit dimensions, to support knowledge management activities (Polanyi, 1966; Bloodgood & Salisbury, 2001). The former is achieved by codified approach. (Gray, 2001; 2003; Zack, 1999). The latter put emphasize on dialoging via social networks to facilitate knowledge sharing. (Agostini et al., 2003; Brown & Duguid, 1991). KMS employs Information Technologies (IT), such as document management and data mining to facilitate the access, reuse and sharing of knowledge assets within and across organizations

2.2 Task-based Knowledge Management

Generally, the operations of enterprises are mainly planned around tasks. For complex tasks, knowledge workers may need to work out problems collaboratively. As knowledge is embedded in task execution, providing task relevant knowledge (documents) to fit the information needs of task execution is important to support effective knowledge management. Knowledge retrieval is considered as a core component in support of performing knowledge-intensive task in business environments (Fenstermacher, 2002; Gartner Group, 1999). Recently, information retrieval (IR) technique has been considered in workflow management systems to assist knowledge workers find task relevant knowledge. Furthermore, information retrieval (IR) technique has been considered coupling with workflow management systems (WfMS) to proactive deliver task-specific knowledge to users (Abecker et al., 2000a; Fenstermacher & Marlow, 1999).

For complex and knowledge-intensive tasks, effective knowledge support requires the collaboration among knowledge workers. Sharing knowledge with peer groups is important in knowledge management (Fischer & Ostwals, 2001). The Computer-Supported Cooperative Work (CSCW) (Rodden, 1991) and Recommender Systems (Resnick & Varian, 1997) also shade additional light on collaboration. CSCW emphasizes on the power of computer system to help group of people perform the tasks in a shared environment (Ellis, 1991; Rodden, 1991). Recommnder Systems employ content-based filtering and collaborative filtering to recommend web pages, movies, books and so on (Goldberg et al., 1992; Pazzani, 1999; Resnick et al., 1997; Schafer et al., 2000).

Accordingly, the modern KMS focuses on enhancing the system knowledge retrieval capability and further promoting knowledge sharing among project-based or peer groups to achieve the objective of KM.

2.3 Information Retrieval and Information Filtering

Information retrieval (IR) deals with the representation, storage, organization of, and access to information items (Baeza-Yates & Ribeiro-Neto, 1999). In the past twenty years, Information Retrieval technology mainly focused on two dimensions: indexing and searching from a large number of documents. Document pre-processing is important before indexing documents, including term transformation and term weighting (Salton, 1988). Term transformation includes case-folding, stemming and stop word omission or further thesaurus substitution. By the empirical study of Salton (1988), the most effective term-weighting approach considers three factors, which are term frequency, inverse document frequency and normalization. The vector model accomplishes the term weighting work by assigning non-binary term weight to each index term in documents (Baeza-Yates & Ribeiro-Neto, 1999). It is easy to represent documents in the form of vectors in vector space through the document preprocessing. Usually, users accomplish search by conducting query, represented as a set of keywords. The keyword set represents the user information needs. The query vector can be defined as $\vec{q} = \{w_{1,q}, w_{2,q}, \ldots, w_{n,q}\}$ and the document vector can be defined $\vec{d_j} = \{w_{1,d_j}, w_{2,d_j}, \ldots, w_{n,d_j}\}$ where n is the total index terms in the system. Documents can be retrieved and presented to users according to the degree of similarity calculated by cosine similarity measure.

In the past few years, research subjects discussed in the field of Information Retrieval were more general, like Document Classification, Document Categorization, User Modeling, Information Filtering, User Interfaces and so on. The Information Filtering technology is acknowledged to be an effective way to reduce the information overload and provide personalized information (Hanani, Shapira, & Shoval, 2001; Hoashi, Matsumoto, Inoue, and Hashimoto, 2000). Information Filtering stresses on maintaining a user profile, in which the system routes the proper information relevant to the user profile (Baeza-Yates & Ribeiro-Neto, 1999; Shapira, Shoval & Hanani, 1999). Information filtering technology relies on the support of traditional kernel technology of information retrieval, but this technology put more emphasis on methods to maintain and learn user profiles to find relevant documents (Baeza-Yates & Ribeiro-Neto, 1999; Widyantoro, Ioerger, &Yen, 2001).

Information retrieval and information filtering technologies applied in document management system are generally the first pace of knowledge management initiatives, since textual data such as articles, reports, manual, know-how documents and so on are treated as valuable and explicit knowledge within organizations (Hahn & Subramani, 2000; Nonaka, 1994). Information retrieval and information filtering are considered as the core techniques to achieve knowledge retrieval. In addition, information retrieval provides not only text processing technique, but also document classification technology to help organizations collect and process documents to achieve the goal of knowledge reuse (Ruffolo, & Iiritano, 2001; Sebastiani, 1999) With the evolution of information filtering system, it not only reduces the problem of information overloading but also provides relevant and interested information to users to accomplish their tasks. Accordingly, maintaining and learning user profiles is important in modern Knowledge Management Systems.

3. K-Support System: Task-based Knowledge Support System

3.1 Rational to Design Task-based Knowledge Support System

"John is a new worker of a software company. He is assigned to a software development project and is responsible for designing a software component. John is facing the problem to understand the assigned task. He wants to find task-related expert or colleague to solve the encountered problem or guide him to the right direction while developing the software component. Unfortunately, workers who have relevant knowledge are busy for the business projects. Hence, John comes up with the idea to find the possible solutions from the document management system or information repository in the organization. However, tremendous amount of data frustrated John. It is hard for John to have a clear view of information structure of the document management system or information repository."

From the above given example, we can understand that there are always problems while conducting a task, especially for novices. When a worker in an organization has problems or information needs while executing a task, he/her might need the knowledge support to help he/her accomplish the execution task. Accordingly, the worker may seek someone who has met this problem or has done similar tasks before. The worker may also try to find the relevant codified knowledge within the organizational repository. Thus, if knowledge resources in an organization are organized via the view of tasks, workers could get help from other workers and save a lot time to find out needed codified knowledge. Consequently, task-based knowledge management systems are helpful to support and share task-relevant knowledge.

Nowadays, many Knowledge Management Systems (KMS) adopted document management systems to help workers search tremendous amount of texts. Most of these document management systems relied on search engines or browsers. Although some KMS adopt subjects' taxonomy to assist workers search organizational repository, it is still difficult for workers to have a gist view about the taxonomy of codified knowledge. In this work, we build a task-based knowledge support system. The proposed system not only can help workers access knowledge resources of organizations in a systematical way, but also help workers find the proper experts or task-based peer group effectively.

How can we achieve task-based knowledge support system? First, we should find out knowledge resources in the organization and organize them based on their characteristics. Workers of our system need to create knowledge resources through their task-based workspaces. The system maintains an information structure to categorize task-based knowledge items. Thus, a worker could get aids based on this information structure when he faces problems.

The proposed system also provides an automatic knowledge support service by implementing a profiling mechanism to generate work/task profiles. A relevance feedback technique is adopted to modify work/task profiles. The workers can find a proper set of task-relevant knowledge while engaging specific task according to his/her work/task profiles. Our system collects feedbacks of each worker. Peer-group based knowledge sharing can thus be supported by analyzing the similarity of interests between workers.

3.2 Concept of Task-based Organizational Environment

Two types of knowledge properties are supported in our task-based workspace: explicit ones and tacit ones. The explicit ones include documents, tasks and categories. Information Retrieval techniques are adopted to extract the knowledge patterns from text documents. The tacit knowledge comes from human resources. The human resources in our system are workers who have conducted several knowledge-intensive tasks. Because their experiences cannot be easily extracted out, they just can be conceptually represented. In addition, the system may locate someone with task relevant knowledge or similar task experience, providing a communication portal of knowledge sharing among workers. The patterns of explicit and tacit knowledge sources can be supported by the aid of profiles. The profiles will be kept in system to support knowledge sharing and retrieval.

The above concepts can be organized from the aspect of codified knowledge and humane resource knowledge. The organization can achieve business goal with proper management of organizational knowledge. Fig. 3.1 illustrats the concepts of our proposed task-based organizational environment.

(1) Knowledge Resource

For executing tasks, workers must seek relevant information to solve problems. Information may come from the workers who have similar experiences, documents about similar topics to their tasks in the organization or other resources. But even if the relevant information is kept in the organization, it is hard to obtain the right information without wasting lots of time to browse large amount of information.

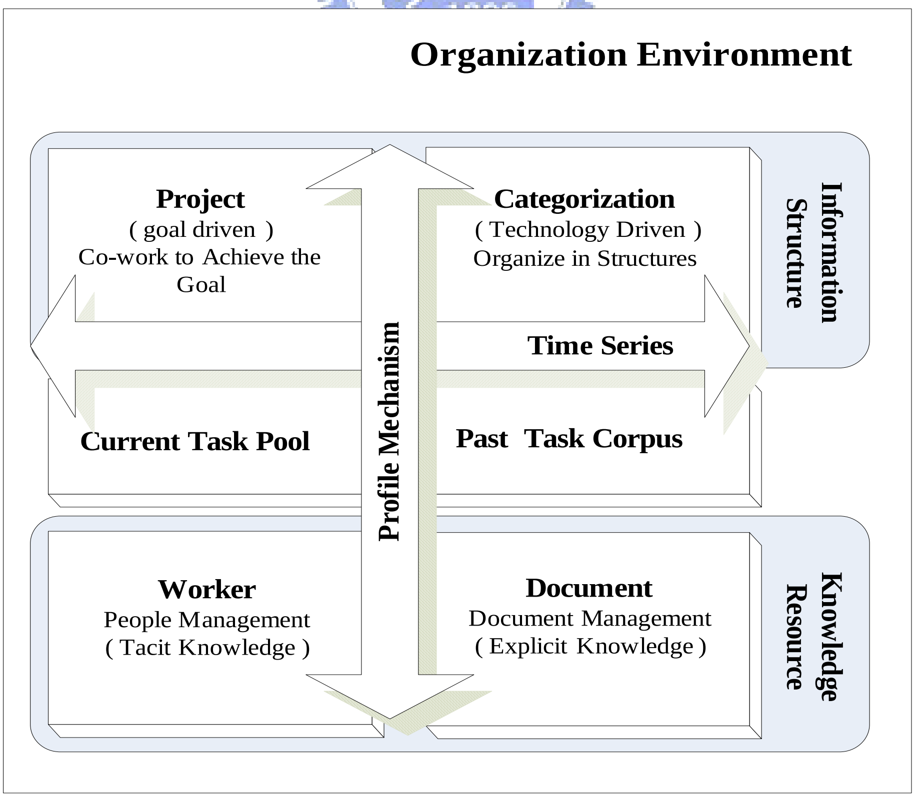

Fig. 3.1 Concept of Task-based Organizational Environment

Under such situation, a catalog of documents could be supported to help workers search the relevant information. If the worker couldn't comprehend with his job, what can help him except for huge amounts of documents? From the task point of view, documents are just a piece of jigsaws to solve some problems in the jobs. One cannot just see a document to understand the whole picture of task. To accomplish a task, various experiences are needed. So, the information or knowledge of past tasks is important resources to support task execution. If we can preserve not only documents but also the past information about tasks, including relevant people and relationships to resources, in the knowledge repository, we can help the knowledge workers get some conceptual information or knowledge from the organizational memory.

(2) Managing Codified Knowledge－From Document to Domain Information Structure

There are numerous tasks carried out in the organization and a huge amount of documents are generated and retrieved during task executions. We design a categorization method to mange document-based knowledge. The main research categories are defined as the domain information structure in the organization. Our system analyzes the relevance of tasks to each research category. Relevant tasks can thus be provided through the domain information structure to facilitate effective access of knowledge resources.

(3) Managing Humane Knowledge－From Knowledge Workers to Projects

The experts play an important role in this framework to define tasks and specify the main domains of the organization. The experts may be the past task executors or domain experts who are familiar with some skills. Meanwhile, there are many projects running in the organization. The project leaders are usually experienced in their professional domains. They are also domain experts. Workers in the same project are cooperating to achieve the business goal.

(4) Managing Knowledge from Task Perspective

Knowledge resources are organized from the perspective of task via putting the tasks in the middle of projects and knowledge resources. The system can locate relevant tasks as well as the needed knowledge of the executing task such as needed background knowledge, technique skills, documents and so on. Our objective is not only to locate relevant documents but also relevant tasks.

(5) The Role of Profile

An assessment procedure is proposed to model the information needs of task workers (profiles). System can collect the worker's feedbacks about tasks to modify the profile. The worker profiles are the kernel to support the tacit knowledge resources and explicit knowledge resources.

3.3 System Overview

The K-Support system is a multi-layered knowledge support system. This system is aiming at enhancing knowledge reuse and sharing by the aid of ITs. Four layers are identified in the proposed system, including *knowledge items collection layer*, *knowledge acquisition layer*, *knowledge modeling layer*, and *front-end application layer*.

***Knowledge items collection layer*:** Large amount of information are generated and accessed during task execution to support workers accomplish task objectives. Therefore, identifying valuable knowledge existing in various forms from different knowledge sources within organization is required. The goal of this layer is to collect codified knowledge and human resource knowledge spread around the organization. The unstructured and semi-structured information embedded in documents such as documents, presentation slides, forms, lesson-learned, database entries, and so on are treated as valuable knowledge items. As we know, effectively codifying tacit knowledge such as domain experts' knowledge is not an easy task; therefore, the system may locate someone with knowledge, providing a communication portal of knowledge sharing among workers.

***Knowledge acquisition layer*:** The knowledge acquisition layer underpins the system to handle and process task-relevant knowledge items. Intellectual content can be

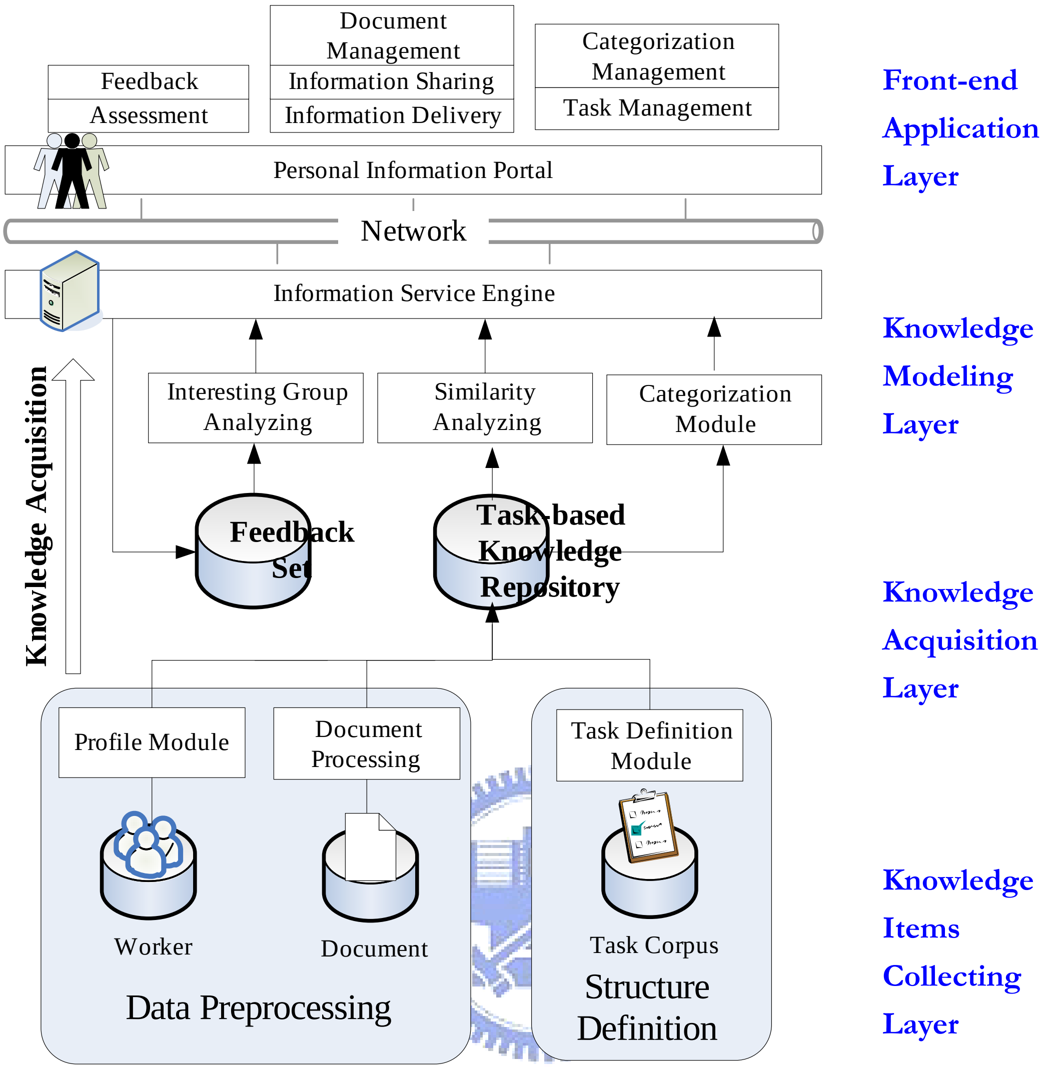

Fig. 3.2 K-Support Knowledge Support System Architecture

codified into explicit form to enhance knowledge reuse and sharing. Two appropriate modules are responsible for processing textual data, especially documents. The basic Information Retrieval and Information Extraction techniques are applied in both modules to process and organize textual data and task-relevant information. All processed task-relevant information is stored in the task-based knowledge repository for sharing, access and reuse.

Knowledge modeling layer**:** To bridge the gap between users and the system, this layer concerns about modeling task information and user information needs to realize task-based intelligent service. The intelligent service, especially the task-based knowledge retrieval service, is based on the task profile modeling approach from workers' view. The workers can find a proper set of knowledge such as documents, related research and humane resource while engaging specific task. On the other hand, the system can infer and present the worker's information needs based on the profile. In addition, the domain ontology is used as a formal representation of the related concepts/objects based on the aspect of business task.

***Web-based GUI and Front-end application layer*:** An integrated platform is build upon *knowledge modeling layer* to construct the task-based knowledge support portal. Different knowledge management tools are available for different user groups; therefore, different task-based portals are created based on the workers' task characteristics. With a task assessment editor, an executor can conduct task assessment to create his/her own task profile. The profile will be generated based on the assessment of codified knowledge and task-relevant knowledge. With a document management editor or categorization management editor, the system administrator can activate the document processing module and categorization module to process task-relevant information. In addition, workers can enter the task-based workspace to access, manage and sharing task-relevant knowledge.

We will test our proposed system architecture in a research environment. The experience of developing the web-based knowledge management system will be introduced and discussed in the remainder sections.

4. Knowledge Support

For facilitating task-based knowledge support, the task profile of execution task is the kernel of the proposed system to discover relevant information. We represent the work profile in terms of topicalities in information research ontology in our research domain. Section 4.1 introduces the domain information structure. The information structure is constructed based on analyzing the task characteristics, task similarity, and task research area, which will be addressed in Section 4.2. The task profile describes the key content of executing-task that the worker conducts at hand. The work/task profile is deployed in the system to provide task-based information services, such as knowledge retrieval and knowledge sharing, which will be addressed in Section 4.3 and Section 4.4, respectively.

4.1 Domain Information Structure Formalization

Managing knowledge contained in documents is an important issue in the knowledge management system. In this section, we describe the essential processes to formalize domain information structure. First, textual data are collected and stored from various sources during executions of each task and then knowledge embodied in the textual data will be extracted by content analysis; second, employing the fuzzy classification method to categorize tasks into categories; third, grouping the tasks based on the result of task categorization. Fig. 4.1 shows the overview of domain information structure.

- **Documents:** Collecting and storing textual data from various sources during each task execution and then extracting knowledge embodied in textual data by content analysis.

- **Tasks:** All incoming textual data are organized in a chronological order from the aspect of task. Tasks are divided into past tasks, namely existing-task set and ongoing tasks, namely executing tasks.

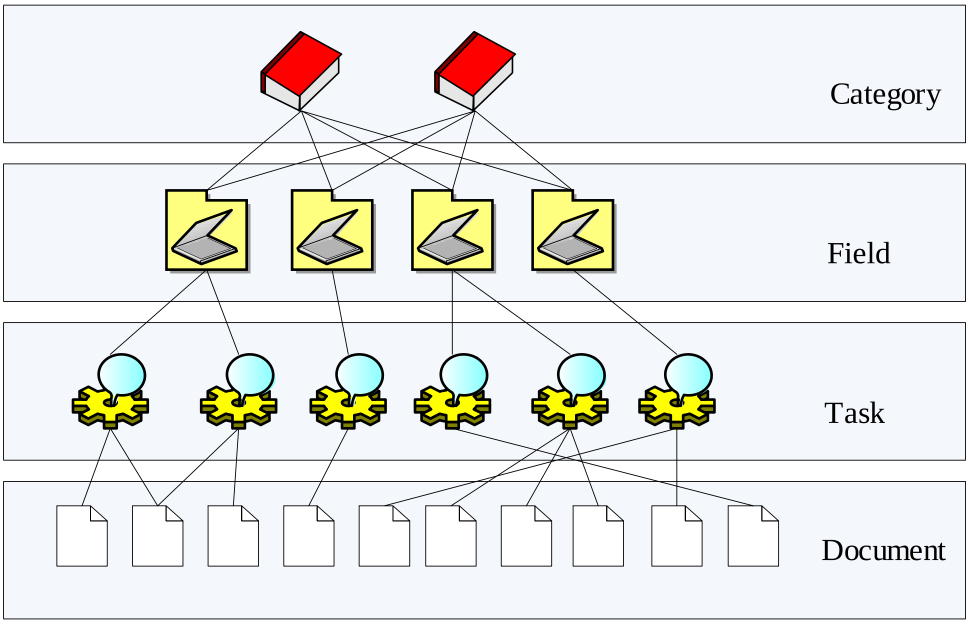

Fig. 4.1 Information Structure Overview

- **Fields:** Many tasks within the organization may have similarities in some aspects. We analyzing the task similarity based on the concepts of each task; namely, their relationships with respect to each category.
- **Categories:** The categories are pre-defined by the experts of target domain. Namely, categories represent the main research area of our target domain.

4.2 Procedure of Generating the Domain Information Structure

In the following, we briefly introduced the overall procedure to constructing the domain information structure. The detailed procedure of constructing domain information structure is presented in Ref. (Wu & Liu, 2003). Some modifications are aroused to meet our target environment while conducting the empirical experiment, such as the term weighting factors, task categorizing methods.

4.2.1 Document Processing

Information retrieval technology is used to extract knowledge items from documents. A document is represented as a vector of weighted terms. Before representing each document in a *n*-dimensional vector space model, case folding, stemming, and stop word removing are adopted in our *document management sub-system* for word transformation. We use the Porter's stemming algorithm (Porter, 1980) and a stop word list in Smart IR system (Salton, 1971) to do the above steps.

Next, we employ the most effective term-weighting approach suggested by Salton (1988) to achieve the term weighting. The three factors in term weighting are term frequency, inverse document frequency and normalization. Therefore, the document is represented as a vector of weighted terms. The weight of a keyword kw_i in a document d is defined as term frequency $f_{kw_i}^d$ multiplied by the inverse document frequency idf_{kw_i}. The following equation shows the vector of document $\vec{\mathbf{d}}$.

$$\vec{\mathbf{d}} = \left\{ w_{kw_i}^d \Big/ \sum_{vector\ d} \sqrt{(w_{kw_i}^d)^2} \;\middle|\; w_{kw_i}^d = (0.5 + \frac{0.5 \times f_{kw_i}^d}{\max f_{kw_i}^d}) \times (\log \frac{N}{f_{kw_i}} + 1) \right\}$$

Notably, we adopted the normalized term frequency during term weighting. According to the empirical experiments of Salton (1988), if there are many technical vocabulary in the data collection and the vocabulary is not very varied, normalized term frequency is a proper term weighting component.

$$w_{kw_i}^d = (0.5 + \frac{0.5 \times f_{kw_i}^d}{\max f_{kw_i}^d}) \times (\log \frac{N}{f_{kw_i}} + 1)$$

where $f_{kw_i}^d$ denotes term frequency of kw_i in document d and f_{kw_i} denotes the number of documents containing kw_i. N is the total number of documents.

4.2.2 Configuration of Categories and Tasks

The vectors of task's kernel documents are aggregated to form the task corpus. Several tasks' corpora can be aggregated to generate a set of terms representing the category. The representative tasks of each category are specified by experts, as shown in Fig. 4.2.

The representative tasks are like the seeds of the category. We apply a fuzzy categorization method to compute the relationships between tasks and categories in our system.

Definition I: Let C be a set of categories, $C = \{C_1, C_2, \cdots, C_m\}$, and let T be a set of tasks, $T = \{T_1, T_2, \cdots, T_k\}$. A fuzzy relation matrix $\boldsymbol{R}$ is shown as follows:

$$\boldsymbol{R} = \begin{array}{c} \\ C_1 \\ C_2 \\ \vdots \\ C_m \end{array} \begin{array}{c} \begin{array}{cccc} T_1 & T_2 & & T_k \end{array} \\ \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1k} \\ g_{21} & g_{22} & \cdots & g_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ g_{m1} & g_{m2} & \cdots & g_{mk} \end{bmatrix} \end{array}$$

Where relevant degree of tasks and categories are calculated by the following equation.

$$g = \mu_i(T_r) = \frac{\sum_{j=1}^{n} \eta_{rj} \varphi_j p_{ij}}{\sum_{j=1}^{n} \eta_{rj}}$$

$$where \quad \eta_{rj} = \begin{cases} 1 & T_r \text{ contain keyword } k_j \\ 0 & otherwise \end{cases}$$

$$\varphi_j = 1 - \left| w_j^{c_i} - w_j^{t_r} \right|$$

Where p_{ij} is the distribution probability that the *j*th keyword will appear in the *i*th category. $w_j^{c_i}$ is the weight of keyword *j* in category i and $w_j^{t_r}$ is the weight of keyword *j* in task *r*. Therefore, φ_j is relative weight of keyword j in category *i* and task *r*. And *r* is the *r*th task, *j* is the *j*th keyword , *n* is the number of keywords in a category.

C1 [Business Intelligent]

T16 → Mining Association Rules for Information Recommendation in Enterprises

T22 →Towards a Framework for Discovering Project-Based Knowledge Maps

T31 →A Collaborative Relevance Feedback Approach to Task-driven Recommendation

C2 [Data Warehousing]

T15 →Deploying a Data Warehouse System for Computer Integrated Manufacturing

T17 →Integrating Data Warehousing and Data Mining for Web Logs Analysis

T27 →Event Detection and Tracking based on event hierarchy

C3 [IS Security]

T4 →A Study on Internet Healthcare Commerce: Integration of Healthcare EDI and SET

T12 →An Implementation of Authorization Management in Task- Based Access Control Models

C4 [Workflow Systems & e-Service]

T7 →Workflow Modeling Based on XML and Rules

T20 →Coordinating Inter-organizational Workflows based on Process-Views

T23 →A Integrated Framework for Recommending Composite e-Services based on Mining and Collaborative Filtering

C5 [Internet Commerce]

T5 →A Flexible Architecture for Intermediary Based Electronic Commerce

T10 →Design and Implementation of XML Based Electronic Catalogs

T25 →A Implementation and Comparison of Collaborative Filtering for Recommendations

Fig. 4.2 Five Categories with Their Representative Tasks

(＊Note: C denotes category and T denotes task.)

4.2.3 Fields Generation

Based on the fuzzy relationship, similar tasks with similarity greater than a specified threshold are grouping together to form a field. A field-to-task matrix can be created to record the memberships (0 or 1) of tasks to fields.

Tasks in the same field represent similar background of organization domains. Fields are used to support information sharing between workers whose tasks are similar. The details are described in ref. (Wu et al., 2004). In the following, we give an example to explain how to generate domain fields.

Example of field generation: Assume there are five pre-defined categories and six existing tasks in our research domain, $C=\{C_1,C_2,C_3,C_4,C_5\}$ and $T=\{T_1,T_2,T_3,T_4,T_5,T_6\}$. The fuzzy relationship between tasks and categories is shown in Table 4.1. If we define a threshold θ to transform the original fuzzy

Table4.1. Example of Fuzzy Relationship between Task and Categories

	C_1	C_2	C_3	C_4	C_5
T_1	0.328369	0.302254	0.264280	0.301422	0.317244
T_2	0.320513	0.368627	0.271819	0.273724	0.403751
T_3	0.226324	0.223173	0.229626	0	0.538180
T_4	0.238224	0	0.686292	0	0.303400
T_5	0	0	0.259265	0.217489	0.599362
T_6	0.272681	0.415553	0.224740	0.223464	0.279570
T_7	0.226185	0	0.26138	0.475512	0.302893
T_8	0.295071	0.26953	0.307478	0.270696	0.296543
T_9	0.252462	0.220751	0.237283	0.221925	0.561252
T_{10}	0.222512	0	0	0.22048	0.593153

Table4.2. Example of Binary Relationship between Task and Categories

	C1	C2	C3	C4	C5
T1	1	1	0	0	1
T2	1	1	0	0	1
T3	0	0	0	0	1
T4	0	0	1	0	1
T5	0	0	0	0	1
T6	0	1	0	0	0
T7	0	0	0	1	1
T8	0	0	1	0	0
T9	0	0	0	0	1
T10	0	0	0	0	1

relation matrix into a binary relationship matrix, the value above the threshold is set to "1" whereas the value below the threshold is set to "0". Several methods can be adopted to set the threshold. We employed min-max operation in fuzzy theory to find out a threshold. The benefit of this method is to examine the relationship of tasks and categories. In our example the threshold is "0.301422". The result is shown in Table 4.2. Finally, the six fields are identified. The fields will be named according to the characteristics of similar task set.

{T1,T2}, {T3,T5,T9,T10}, {T4 },{T6},{T7}{T8} →Field={F1, F2, F3, F4,F5,F6}

4.3 Profile Modeling based on Domain Information Structure

As mentioned previously, past task information is helpful for constructing the concept of new tasks. An assessment procedure, *two-phase collaborative relevance assessment approach*, is implemented to model the information needs of task workers (Wu & Liu, 2003). From the assigned task definition and the worker's cognition to his/her task, we can support the past similar tasks to support the worker. Furthermore, the profile can be adapted by the Relevance Feedback algorithm. Namely, the system can infer the information need about a specific task based on the user's feedback. And further modify the profile based on feedback analysis.

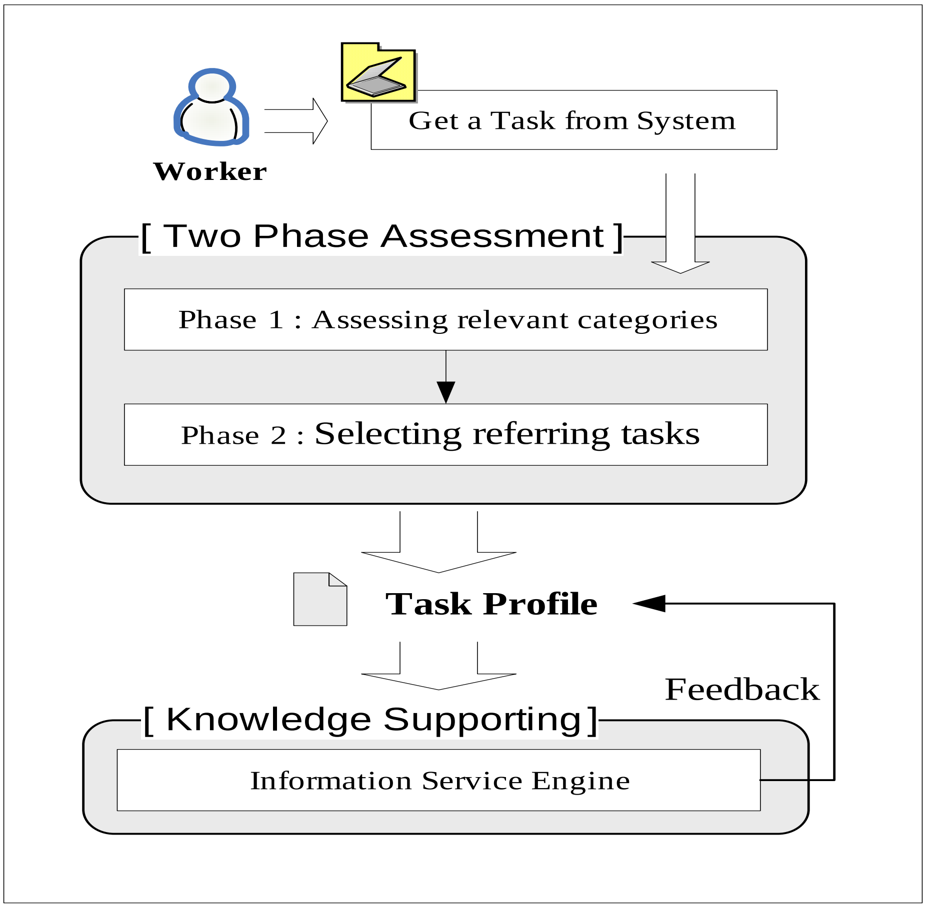

Fig. 4.3 Profile Modeling Procedure to Support Information Services

4.3.1 Generating New Worker's Task Profile

Our system need to know how the workers conceive with their tasks, especially for novice workers. Therefore, we implement an assessment procedure to model the workers' task profiles. The procedure can help the worker construct the concept of his task. The features of systematic assessment approach－*two-phase collaborative relevance-assessment approach* are:

(1) Considering the relevance degree while conducting relevance assessment.

(2) Systematic assessment with the support of domain information structure.

(3) Collaborative assessment on the relevance of existing tasks via collaboration of cooperative workers and domain experts.

When a worker executes a new task, the system will generate an initial profile based on the task definition and the analyzing historical relevant tasks in the organizational repository. The two-phase assessment is briefly summarized as follows.

Phase1－Assessing relevant categories. According to our empirical study, the novices cannot make a proper category assessment without a gist view about the research area. Therefore, providing collaborative mechanism in the system is important for novices.

Phase2－Selecting referring tasks. After completing phase one, the system filters out a set of irrelevant tasks. Then the system shows the referring tasks for workers to conduct further task assessment.

After finishing the assessment, the system will modify the initial profile based on the worker's assessment to form a task-worker profile and be stored in the system. The task-worker profile is generated by the equation in the below, which is modified from Rocchio's (1971) and Ide_Dec_Hi's (1971) Algorithms.

$$\vec{\mathbf{S}}_{new} = \alpha \vec{\mathbf{S}}_{initial} + \beta \sum_{\forall t_j \in T_r} (w_{t_j}) \vec{\mathbf{p}}_{tj} - \gamma \max_{2-non-relevant} \sum_{\forall t_j \in T_n} (1 - w_{t_j}) \vec{\mathbf{p}}_{tj}$$

Where,

$\vec{\mathbf{S}}_{new}$ Represent a new task profile of a specific execution task which is derived from initial task description and relevant task corpus.

$\vec{\mathbf{S}}_{initial}$ A initial task profile derived from initial task description

Tr, Tn Set of relevant and irrelevant tasks

$\vec{\mathbf{p}}_{t_j}$ The corpus of task t_j

w_{t_j} The relevance degree of the task t_j to the executing task.

The Fig. 4.4 is an example of task profile. The second column of **pserial** denotes task executor "P0000000026" and the third column of **tserial** denotes "T0000000029" an executing task. The weight is computed by the method described in Section 4.2.1.

prid	pserial	tserial	keyword	weight	date
3603	P0000000026	T0000000029	pseudo-relev	0.0961845	200401010000
3604	P0000000026	T0000000029	result	0.0517687	200401010000
3605	P0000000026	T0000000029	purpos	0.0566134	200401010000
3606	P0000000026	T0000000029	unexpect	0.113227	200401010000
3607	P0000000026	T0000000029	mall	0.226453	200401010000
3608	P0000000026	T0000000029	match	0.0856314	200401010000
3609	P0000000026	T0000000029	measur	0.10209	200401010000
3610	P0000000026	T0000000029	mechan	0.0655984	200401010000
3611	P0000000026	T0000000029	method	0.142884	200401010000
3612	P0000000026	T0000000029	methodolog	0.1863	200401010000

Fig. 4.4 Example of Task Profile

4.3.2 Task Profile Adaptation

The mechanism of profile modeling uses a collaborative and interactive approach. Furthermore, the system can capture the perception of current worker's status and adapt to the changes by the relevance feedback method. Because the profile is modeled based on the information structure of the research domain; therefore, a lot of useful information can support the system to infer the information needs of the work.

4.4 Knowledge Support and Sharing based on Task Profile

All the knowledge sources, including tacit and explicit ones, can be supported based on the profiles. The profiles will be used to support knowledge retrieval and sharing. The detail of achieving knowledge support by the task profile is presented in our working paper (Wu et al., 2004).

Once a worker has finished the assessment, the system generates his/her task profile. The K-Support portal will help workers find out the most relevant information in their workspace. By using cosine similarity function to calculate the similarities between task profiles and documents, the top-5 similar tasks and top-10 similar documents are listed on the information delivery entry of the portal. The workers can browse the supporting information and conduct feedbacks to the recommended items, as shown in Fig. 4.5.

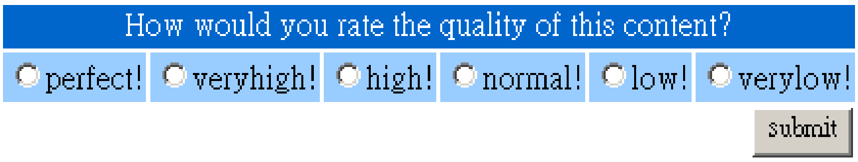

Fig. 4.5 Degree of Relevance Feedback

We adopt the relevance feedback technology to modify the profiles of workers, and record feedbacks of tasks. A matrix, worker-to-task matrix, representing workers' interests on tasks can be generated, as shown in Table 4.3.

The two matrixes, field-to-task matrix and worker-to-task matrix are multiplied to generate the worker-field matrix. The procedure to derive field-to-task matrix is described in Section 4.2. Table 4.4 shows the field-to-task matrix. Table 4.5 shows the multiplication result (worker-to-field matrix).

Table 4.3 The Workers' Rating to Task Set (Worker-to-Task Matrix)

	T1	T4	T16	T17	T18	T22	T23	T24	T25	T26	T27	T30	T31	T50	T51
P26	0	0	0	0.35	0	0	0	0	0	0.65	0.85	0.65	0.35	0	0
P28	0.85	0	0.05	0	0	0.95	0	0	0	0	0	0.05	0	0	0.99
P29	0	0	0	0	0	0	0	0	0	0.6	1	0	0	0	0
P35	0	0	0	0	0	0	0	0	0.98	0	0	0	0	0	0.38
P40	0	0	0	0	0.85	0.75	0.85	0.95	0.65	0	0	0	0	0	0
P43	0	0	0.4	0	0	0.2	0	0	0	0	0.5	0	1	0.6	0
P44	0	0	0.9	0	0.95	0	0.95	0.9	0.95	0	0	0	0	0	0
P46	0	0	0	0	0	0	0	0	0	0.81	0.81	0	0	0	0
P47	0	0	0	0	0	0	0	0	0	0	0	0	0.75	0	0
P48	0	0	0	0	0.85	0	0.85	0.85	0	0	0	0	0	0	0
P49	0	0.28	0	0	0	0.7	0	0	0	0	0	0.7	0.42	0	0.56
P50	0	0	0.85	0	0	0	0	0	0	0.5	0	0	0.99	0	0

Table 4.4 The Field-to Task Relationship Matrix (Field-to-Task Matrix)

	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T30	T31	T50	T51
F1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0
F2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
F3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
F4	0	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0
F5	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F6	0	0	0	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F7	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F8	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0
F9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
F10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0
F11	0	0	1	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
F12	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Table 4.5 The Workers' Rating to Field Set (Worker-to-Field Matrix)

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12
P26	0.35	0.65	0	1.2	0	0	0	0	0	0	0	0.65
P28	1.99	0.9	0	0	0	0	0	0	0	0	0	0
P29	0	0	0	1	0	0	0	0	0	0	0	0.6
P35	0.38	0	0	0	0	0	0	0	0	0	0.98	0
P40	0.75	0	0	0	0	0	0	0.85	0.95	0.85	0.65	0
P43	1.6	0.6	0	0.5	0	0	0	0	0	0	0	0
P44	0.9	0	0	0	0	0	0	0.95	0.9	0.95	0.95	0
P46	0	0	0	0.81	0	0	0	0	0	0	0	0.81
P47	0.75	0	0	0	0	0	0	0	0	0	0	0
P48	0	0	0	0	0	0	0	0.85	0.85	0.85	0	0
P49	1.12	0.7	0.56	0	0	0.28	0	0	0	0	0	0.28
P50	1.84	0	0	0	0	0	0	0	0	0	0	0.5

The relationship of workers to workers can be analyzed to form peer groups. The workers' task-similarities are calculated by cosine measure. Each worker's task-interest can be represented as a feature vector $\vec{P}_i$, as shown in below:

$\vec{P}_{26}$ = <0.35,0.65,0,1.2,0,0,0,0,0,0,0,0.65>

$\vec{P}_{28}$ = <1.99,0.9,0,0,0,0,0,0,0,0,0,0>

$\vec{P}_{29}$ = <0,0,0,1,0,0,0,0,0,0,0,0.6>

The similarity of **P$_{26}$** and **P$_{28}$** is calculated as below

$$Sim(P_{26}, P_{28}) = \frac{\vec{P}_{26} \cdot \vec{P}_{28}}{|\vec{P}_{26}||\vec{P}_{28}|} = \frac{1}{P_{26}P_{28}} \sum_{F=1}^{\dim(field)=l} p_{F,P_{26}} \times p_{F,P_{28}}$$

$\vec{P}_{26}$ and $\vec{P}_{28}$ are worker 26 and worker 28's feedback values which aggregate in field-level, respectively.

The similarity between **P$_{26}$** and **P$_{28}$** is derived as follows,

$$Sim(P_{26}, P_{28}) = \frac{(0.35, 0.65, 0, 1.2, 0, 0, 0, 0, 0, 0, 0, 0.65) \bullet (1.99, 0.9, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)}{\sqrt{(0.35)^2 + (0.65)^2 + (1.2)^2 + (0.65)^2} \times \sqrt{(1.99)^2 + (0.9)^2}} = 0.38$$

The similarity between $\mathbf{P_{26}}$ and $\mathbf{P_{29}}$ is derived as follows,

$$Sim(P_{26},P_{29})=\frac{(0.35,0.65,0,1.2,0,0,0,0,0,0,0,0.65)\bullet(0,0,0,1,0,0,0,0,0,0,0,0.6)}{\sqrt{(0.35)^2+(0.65)^2+(1.2)^2+(0.65)^2}\times\sqrt{(1)^2+(0.6)^2}}=0.88$$

Table 4.6 The Workers Similarity Matrix (Worker-to-Worker Matrix)

	P26	P28	P29	P35	P40	P43	P44	P46	P47	P48	P49	P50
P26	1	0.38	0.88	0.08	0.09	0.56	0.1	0.84	0.23	0	0.45	0.33
P28	-	1	0	0.33	0.37	0.96	0.39	0	0.91	0	0.88	0.88
P29	-	-	1	0	0	0.24	0	0.97	0	0	0.1	0.13
P35	-	-	-	1	0.48	0.32	0.58	0	0.36	0	0.27	0.35
P40	-	-	-	-	1	0.37	0.99	0	0.41	0.84	0.31	0.4
P43	-	-	-	-	-	1	0.39	0.2	0.9	0	0.83	0.87
P44	-	-	-	-	-	-	1	0	0.43	0.78	0.33	0.42
P46	-	-	-	-	-	-	-	1	0	0	0.13	0.19
P47	-	-	-	-	-	-	-	-	1	0	0.75	0.97
P48	-	-	-	-	-	-	-	-	-	1	0	0
P49	-	-	-	-	-	-	-	-	-	-	1	0.78
P50	-	-	-	-	-	-	-	-	-	-	-	1

Notably, two workers may have different feedback on tasks even though those tasks are similar. We raise the feedback level to fields, to further find out the peer groups that do not have similar feedback on the same targets (tasks).

The similarities between P26-P29 and P26-P28 are 0.88 and 0.38 respectively. Obviously, P26 and P29 have higher similarity than P26 and P28. The system will recognize them as the same task peer-group. The system analyze task-peer group based on users' feedback. Finally, we can merge the feedbacks of P26 and P29 and shown in the form of information structure, a tree-like structure is shown in Fig. 4.6. We name the tree-like structure as a sharing tree. The sharing tree reflects the workers' task peer groups. Workers can expand this tree to get further information about the tree nodes. There are many kinds of information on it, including fields, tasks and workers. One worker can further know about the other worker's feedbacks of his interesting nodes and decides if he wants to browse deeply around

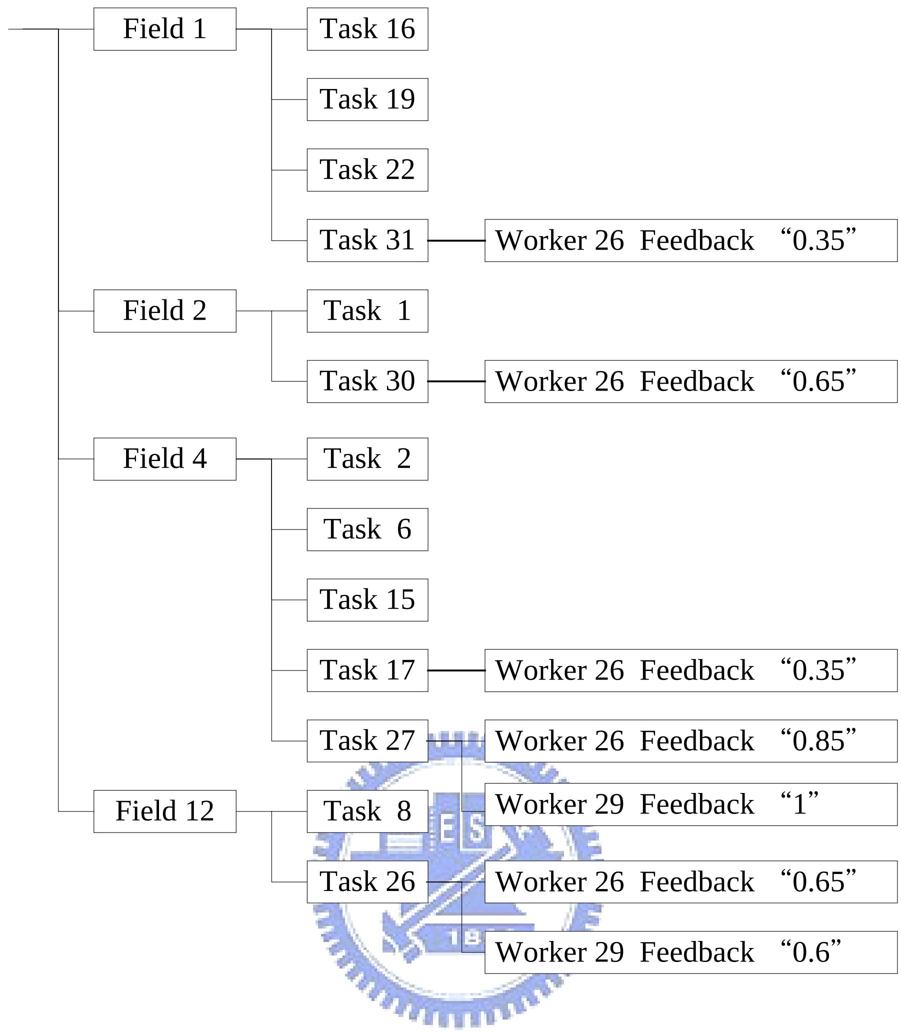

Fig. 4.6The Expanded Task Profile of Worker P29

this tree. Some tasks that are not feedback by any one in the peer group are also shown in the tree, because one of the tasks under the same field has been feedback, and we think other tasks might help the worker. Notably, when new workers are added or someone else conduct assessment of his task or conduct feedback on the recommended knowledge items, this tree may be changed after the system computation. Apparently, the system can locate task peer-group and further promote the sharing activities among task peer-group.

When more and more people make feedbacks or change feedbacks to knowledge items, the proposed system can dynamically calculate peer groups.

5. System Implementation and Development

This work designs and implements a task-based K-Support platform to support knowledge retrieval and sharing. The whole system architecture is divided into layers according to practical implementation. In order to develop an adaptive and flexible system, we employ object-oriented methodology to build each component. Fig. 5.1 shows the layers in the proposed system, including knowledge repository layer, data access layer, service layer, application layer, transport layer and interface.

5.1 Knowledge Repository Layer

This layer is responsible for preserving task-relevant information. All the information is stored in a database. We adopt the MySQL database management system, and install a web administration tool to help administrator manage the task-relevant information. The database schema is shown in Fig. 5.2.

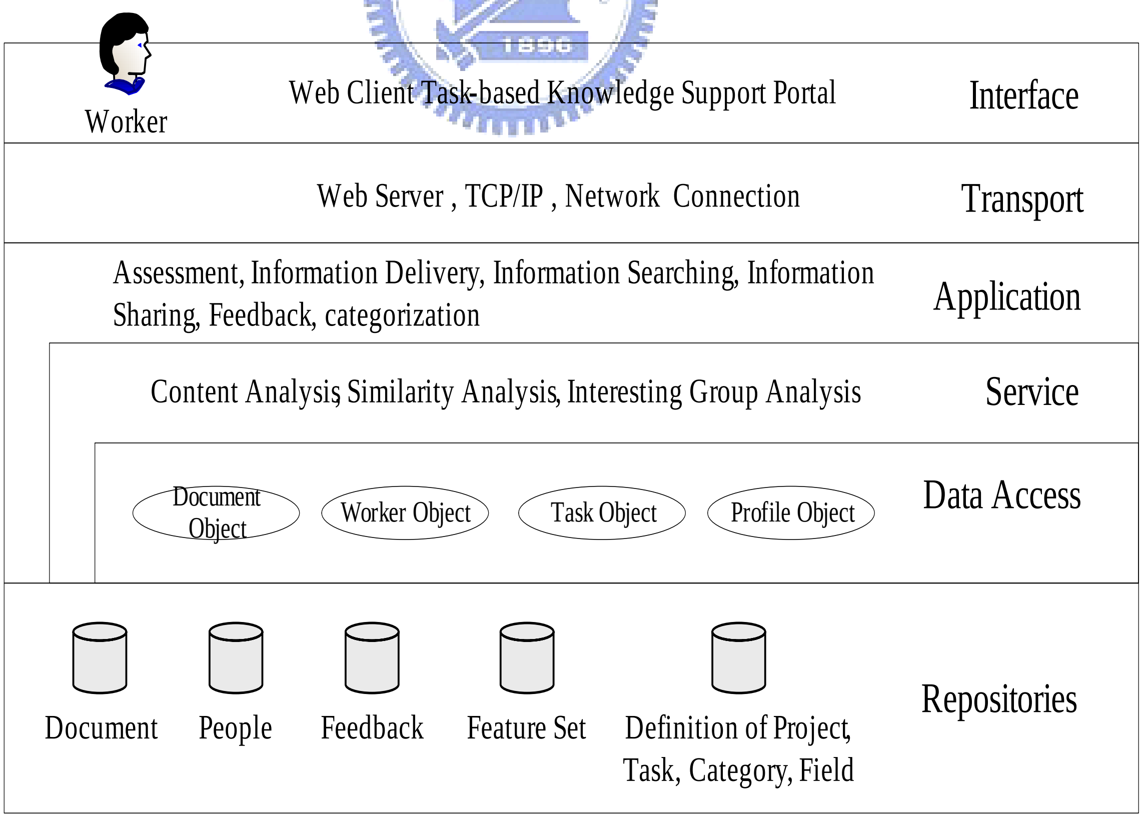

Fig. 5.1 System Implementation Platform

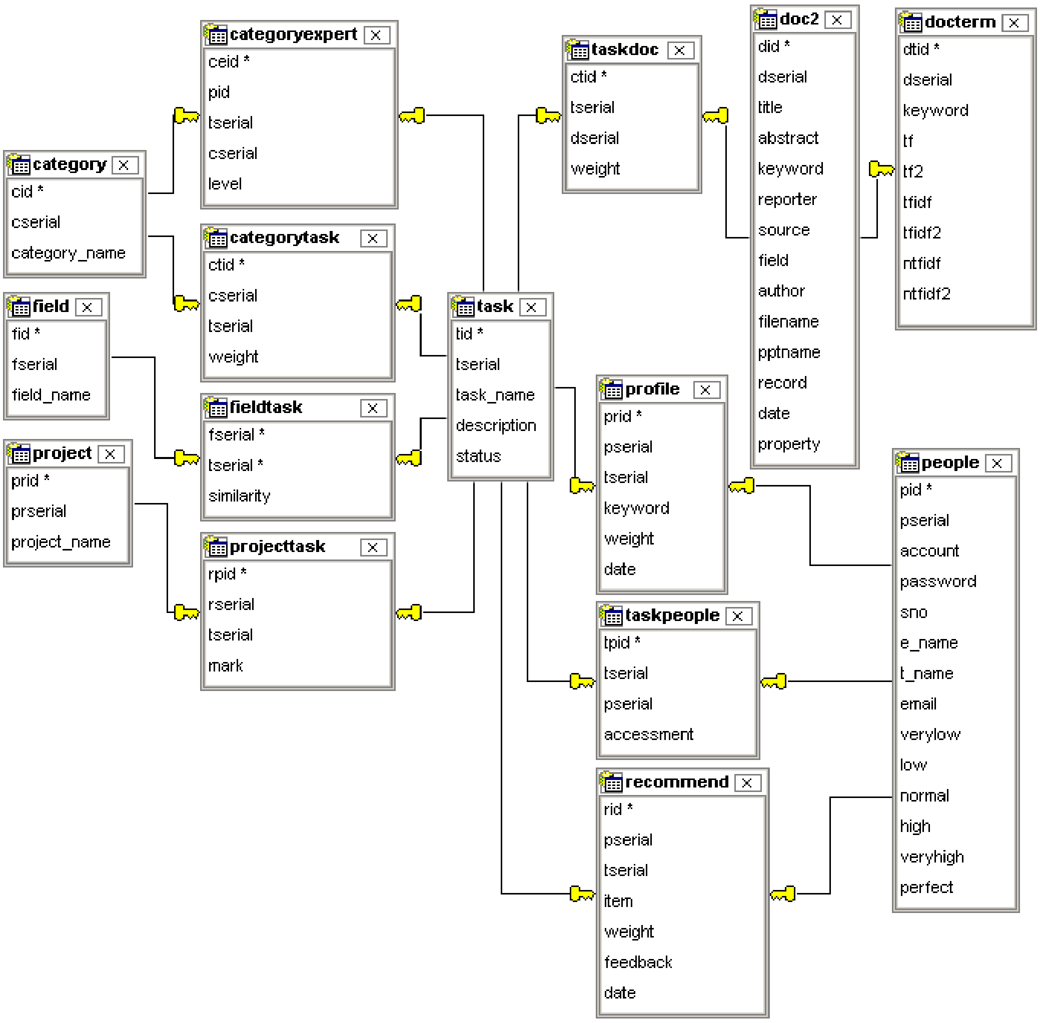

Fig. 5.2 Database Schema in K-Support System

The original documents are stored in the file system and the directory paths of the files are recorded in the database. The information is stored and loaded through the data access layer. When the system activates some functions to conduct data processing, only the needed information will be loaded. Furthermore, the information is treated as the input of relevant objects.

5.2 Data Access Layer

The data access layer fetches data from the knowledge repository layer, and organizes them into objects. This layer is also responsible for creation, modification and deletion of objects, and the translation of object data into the repository. Each type of objects has its related access methods. The knowledge sources are organized in an object-oriented style, thus the proposed system can achieve a flexible layout of knowledge sources and structured data. Four main objects are defined in our system: *document object, worker object, task object* and p*rofile object*.

- **Document object:** A document object contains raw data and the feature set of a document. The raw data includes the slides, texts, and attributes in the document table. The feature set includes terms and weights in the document term table. Different term weighting components are supported in the document object for different situations.
- **Worker object:** A worker object includes the personal information and a set of perception values. The values are divided into six levels, including very low, low, normal, high, very high, and perfect. The worker objects are usually embedded in the document object or task object. For example: in general, a task involves several task executors. The worker object can be embedded into task object to represent the relationship between the task and worker.
- **Task object:** A task object comprises task descriptions and the corresponding relevant information like workers and documents. A task object also contains a feature set regarding its kernel documents and descriptions. The task objects are extracted from current executing tasks and existing tasks. Tasks corpora are categorized into the organization information structure, as introduced in Section 4.2.
- **Profile object:** A profile object contains a feature set and varieties of content analysis methods. The similarity of the objects can be analyzed by the profile object. We can choose different weighting schemes to select a proper set of feature terms to conduct the analysis.

5.3 Service layer

Several information services are performed by manipulating objects developed in the system. These services include content analysis, similarity analysis and peer group analysis. The system provides a certain service by specifying the target objects and the needed service operations.

5.4 Application layer

Several front-end applications are provided by the K-Support system. Applications utilize services and small programs to access knowledge resources. These applications are briefly described in the following:

- **Categorization:** Categorization function is provided by the categorization management application. When a task is finished, this function will be executed to maintain the categorization of task into information structure.

- **Assessment:** Every worker conducts task assessment to generate his task-profile for further usage. A two-phase assessment procedure described in Section 4 is implemented to generate the worker's task profile.

- **Feedback and Information Sharing:** The worker can conduct feedback on recommended items. These feedbacks are stored in the system to derive the peer group. The worker can obtain the task-relevant information of workers with similar task interests (namely, task peer groups), and corresponding relevant knowledge resources (see details in the Section 4).

- **Information Delivery and Searching:** Every knowledge resources in the repository and task corpora (features of tasks) can be supported and searched by the proposed system. When any new information is added into the system, the worker can evaluate if this information is relevant to his task. A simple search function is provided to help user query needed documents, as shown in Fig. 5.3.

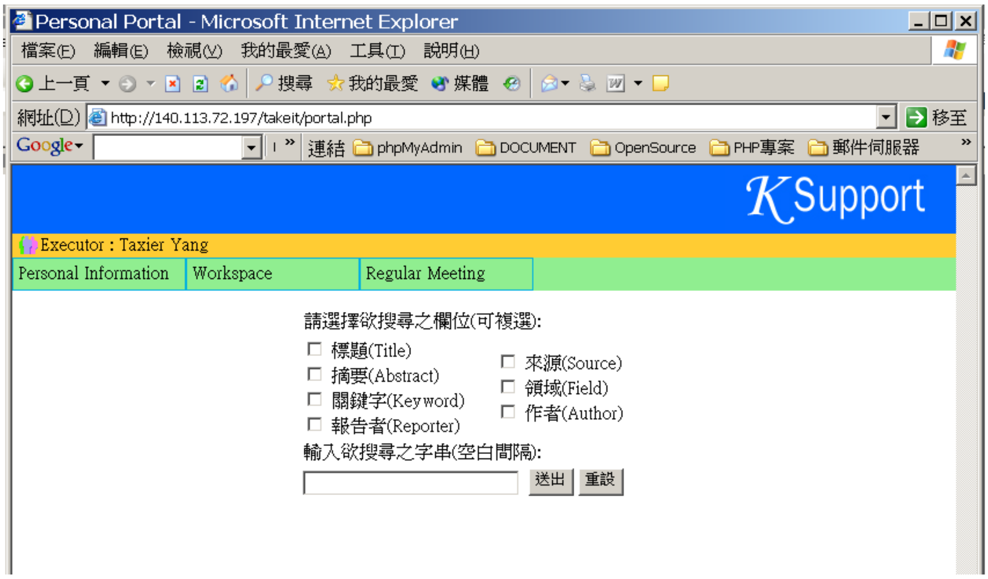

Fig. 5.3 Interface of Search the Organizational Repository

- **Repository Maintaining**

(1) Document uploading and processing: Workers can upload their task relevant sources, including papers, slides or research notes. A web page form is designed to record the relevant attributes of the resources, as shown in Fig. 5.4

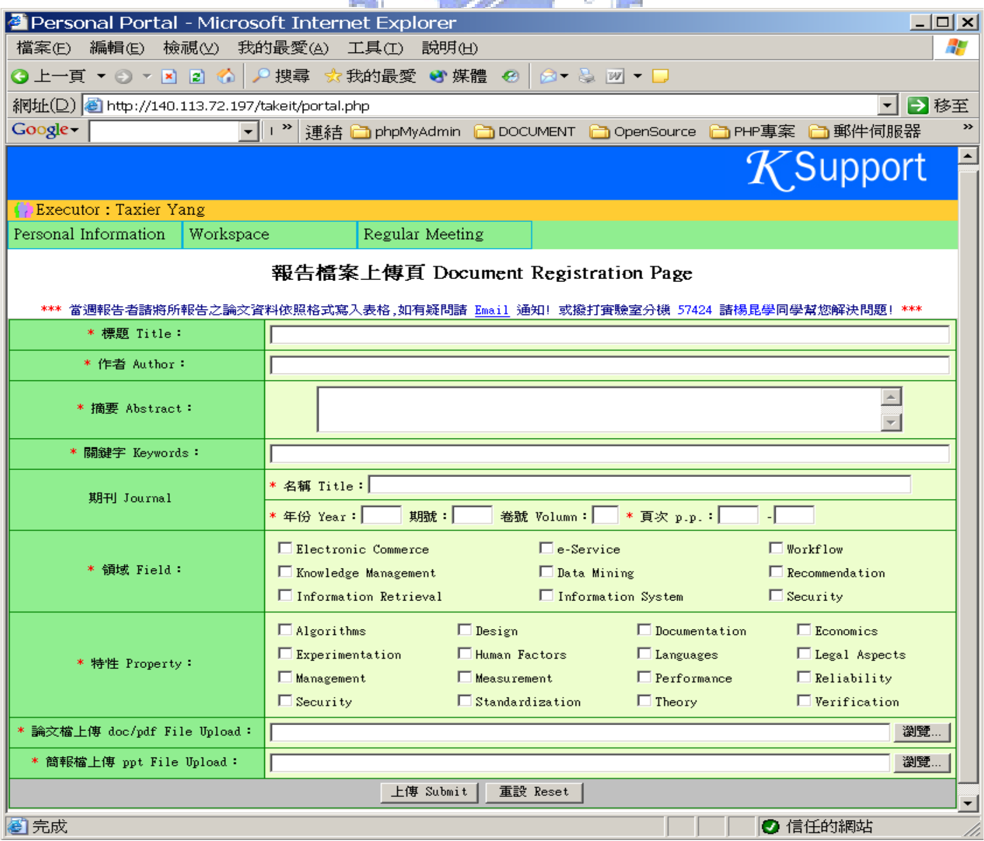

Fig. 5.4 Interface of Document Uploading

When a document is uploaded to the system, it will be marked with D* as its temporal serial number. These documents will be selected to perform document processing step in the management mode of the system. Each document will then be assigned a ten bit long serial number to be stored in the database.

(2) User registration: Workers in our system must register their personal information. Additionally, our system also needs to collect the worker's perception values (see Fig. 5.5). These values are recorded in the system. The system will automatically transfer the worker's feedbacks to specified values of that worker. We also provide an interface to modify the workers information in the management mode.

(3) Task editor: The definition of tasks is the most important part in a task-based system. Experts could ask the system administrator to create a new task into the task pool.

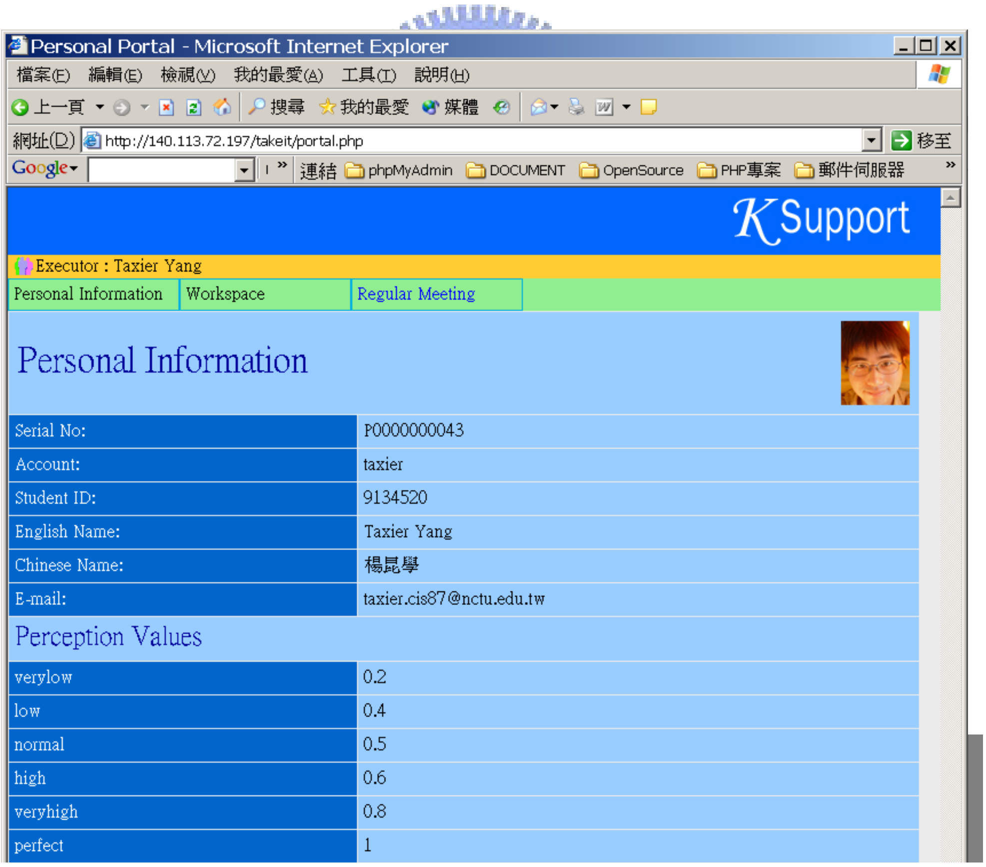

Fig. 5.5 Interface of Worker's Information

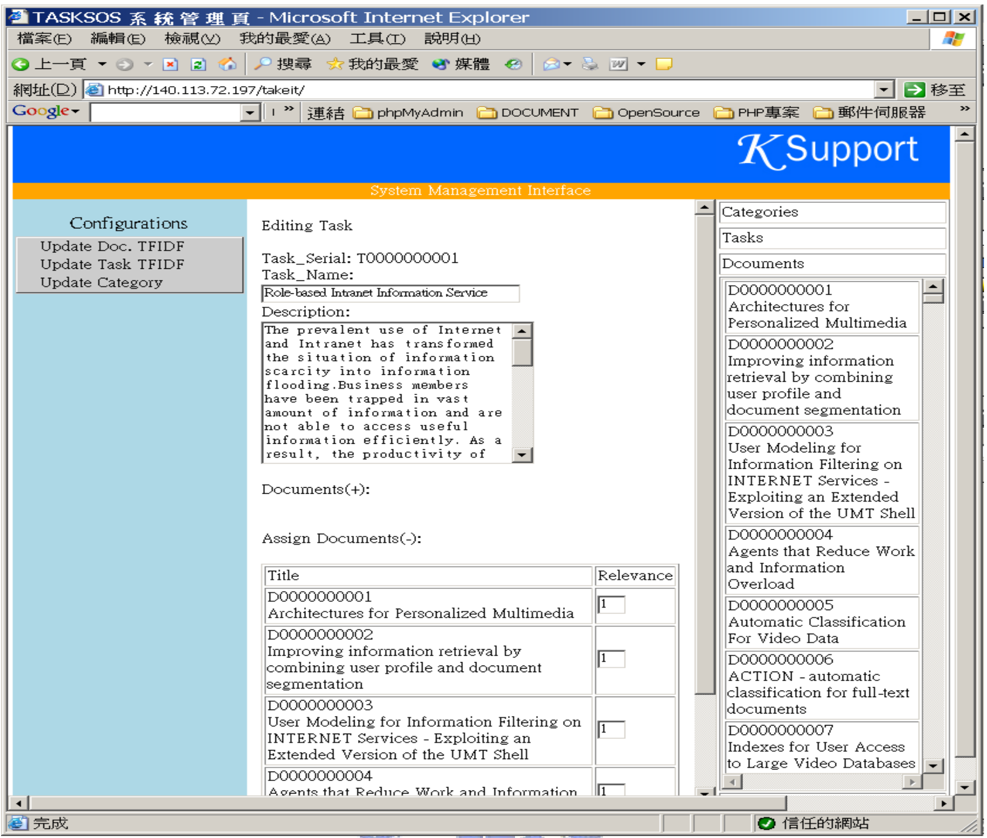

Fig. 5.6 Interface of Task Editor

In the task editing mode, an expert could also browse worker list and documents. The relevant experts or relevant documents can be added to the newly created task. Additionally, the workers who finished his job can use the editor to modify his task information before deriving the task corpus.

5.5 Transport Layer

To provide easy use of our system regardless of hardware limitations and location distances, the system is implemented in a client-server model, in which a web server is used. The server platform is operating on the OS version of Linux Red Hat 9.0, with an Apache web server as the server of the system. Workers can use Intranet or Internet to login into the system via HTTP connections through a simple id verification method integrated with the browser.

5.6 Interface Layer

All the functions are integrated into the task-based knowledge support portal. This portal is developed by using HTML and JavaScript. In such implementation, the system can support various information including texts, sounds and images. An interaction environment is also provided. Workers can obtain relevant information through the browsers of their computers without worrying about the upgrading problem, since all the interfaces can be modified in the server side.

6. System Demonstration

6.1 Data Collection and Participants

6.1.1 Data Collection

The test document set used in our experiment is collected from the laboratory in the institute of information management department. There are five categories to represent main subjects in our experimental domain. They are defined according to the features of the task set based on the schema of ACM Computing Classification Systems (1998). There are 51 tasks in the experimental domain; 31 tasks are existing-tasks and 20 are executing-tasks. The given document set generated or collected from tasks is around 500 documents, which covers the periods of 1996-2004.

6.1.2 User Groups

Two kinds of user group are chosen to conduct the experiments. One group is users who are familiar with the executing task (named experienced users in this work) and another group is users who are not familiar with the executing task (named novice users in this work). Because all are new users in the proposed knowledge support system, we give a short tutorial of using the system.

6.1.3 Task Description

Tasks are divided into existing-tasks and executing tasks:

- **Existing-Task:** When a task is finished, all task-relevant information will be stored in the organizational repository. The kernel documents to support the task execution need to be identified by the workers. Consequently, the task corpus is generated based on the feature selection and weighting methods of standard IR technique.
- **Executing-Task:** The states of tasks conducted at hand are changed according to the situation of the execution. Strictly, a task instance comprises a worker and a definition. Two workers working on the same task will have two task instances.

6.2 System Demonstration

6.2.1 Collaborative Two-phase Task Assessment Scenario

(1) Scenario Description

The worker can obtain help from experts in conducting the assessment. The worker can access a list of referring tasks. He can arbitrarily click for further information about a specific task. Finally, after finishing the assessment, he can enter his task workspace to access the relevant knowledge sources provided by the system.

(2) System Operation Procedure with Associated Interface

Step1: Choosing a Task from System

In the task-based K-Support portal, the worker needs to initially select a task from his workspace. The workspace shows the task information in the order of the joined project. The worker chooses “Task-based Knowledge Recommender System” under the project, “Task-based KM System”. Then the system displays the task information and the items for assessment. The worker can read the task description, relevant documents and the relevant workers also involve in this task. Fig.6.1 shows the interface of the task-based K-Support portal.

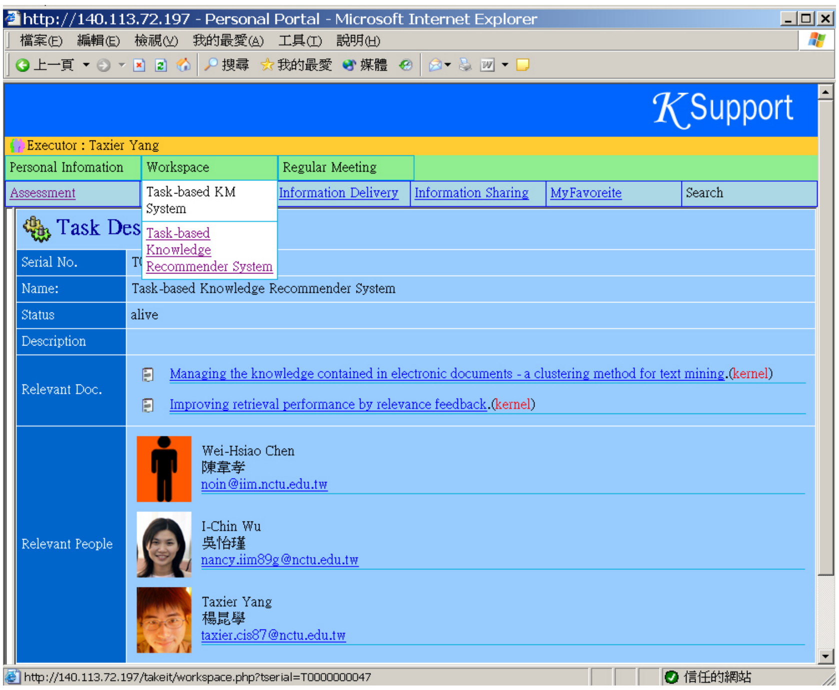

Fig. 6.1 Interface of Task-based K-Support Portal

Step2: Phase 1－Assessing relevant categories

The worker may conduct task-assessment to generate his own task profile. If he selects the "assessment" item, the system will guide him to conduct phase 1 assessment － assessing the relevant categories. The worker should give his perceptions of each category. Besides the worker's perception about the task, he can choose the "expert" column to help him conduct assessment. Additionally, the task information is also shown in below. The worker can look up the information and conduct assessment in the same time. If the worker has done the phase 1, he can click "submit" to enter the phase 2. Fig. 6.2 shows the interface of phase one assessment.

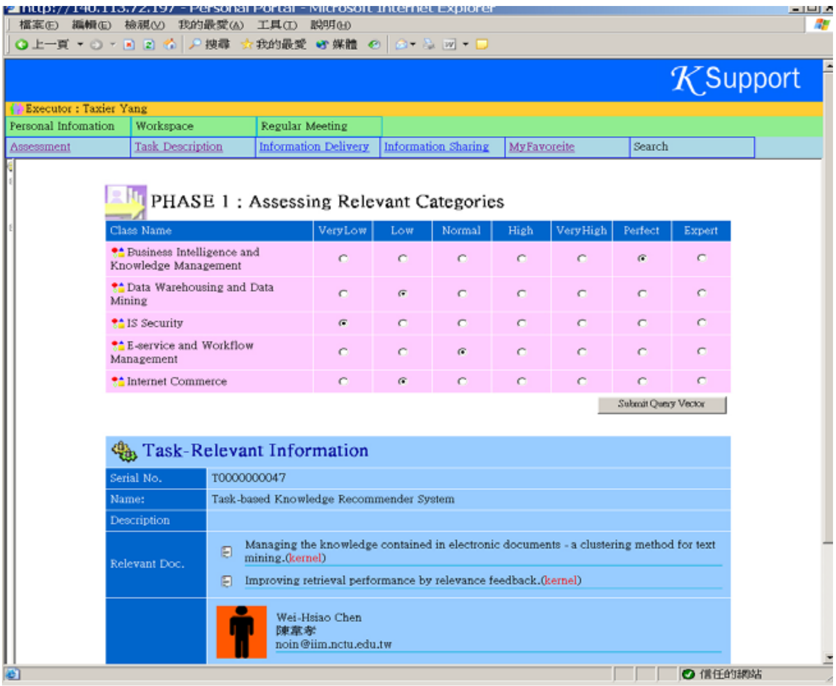

Fig. 6.2 Interface of Phase One Assessment.

Step3: Phase 2－Selecting referring tasks

When the worker enters phase 2, he will see the referring tasks recommended by the system based on the result of phase 1. Referring tasks with associated similarity values to the executing task "Task-based Knowledge Recommender System" will be presented in the system interface. The worker should choose the relevant referring tasks to modify his task profile. After finishing this phase, a personal task profile will be generated. A task profile contains a set of discriminating terms,

$\vec{s}_{new} =< w_{1,new}, w_{2,new}, \cdots, w_{q,new} >$. $w_{i,new}$ stands for the associated weight of ith keywords to task t_{new}. q is the total number of weighted discriminating terms of a task.

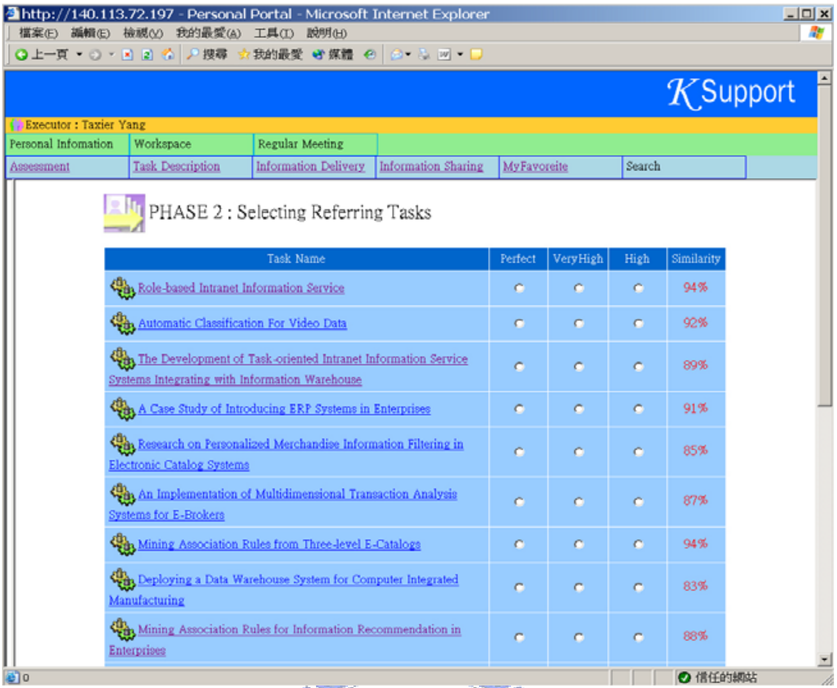

Fig. 6.3 Interface of Phase Two Assessment

6.2.2 Task-based Knowledge Support Scenario

(1) Scenario Description

Everyone who finished the assessment can enter his task workspace. The system will recommend the task-relevant and the latest information based on the task profile Workers could accept or reject these knowledge items by clicking the feedback web form. Meanwhile, the system will receive feedbacks and modify worker's profile. Additionally, once the worker has conducted the assessment or the feedbacks, the system will calculate the worker's task peer group. Therefore, the system can locate members of task peer group for the worker. Consequently, workers can share knowledge within the task peer group.

(2) System Operation Procedure with Associated Interface

- **Information Delivery of System**

Once the worker has finished the assessment procedure, he will receive the information delivered by the system when he log into the system. The first top-5 tasks and top-30 documents are recommended by the system. The documents are retrieved from the repository whereas tasks are retrieved from the existing task set. If the information recommended by the system is not enough, the system also suggests the top-10 relevant keywords to the worker. The worker can search the organizational repository or can search in other heterogeneous repositories. Fig. 6.4 shows the interface of information delivery in task-based workspace.

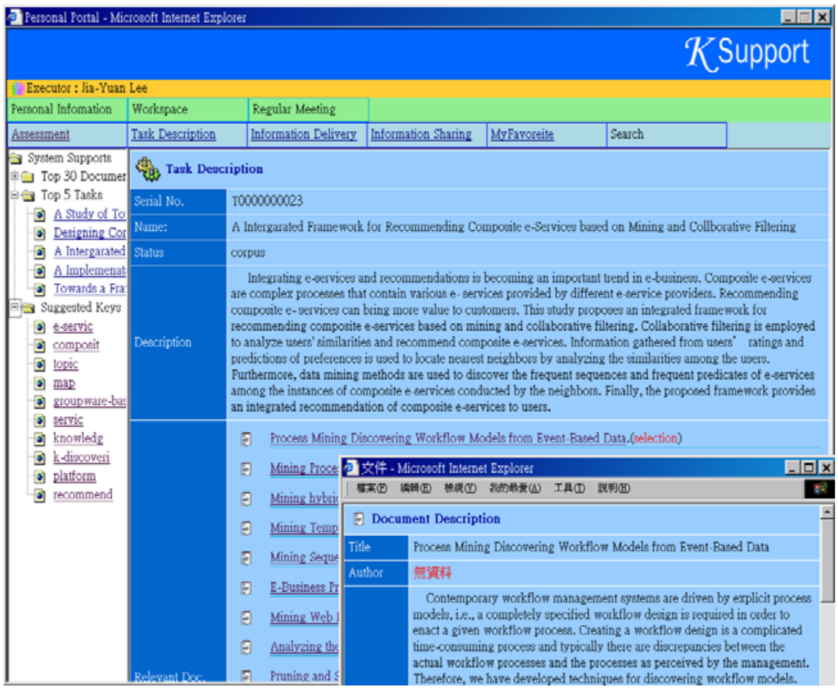

Fig. 6.4 Interface of Information Delivery in Task-based Workspace

- **Feedback Form**

The worker can conduct feedback through the provided interface. Six relevant degrees are provided by the system- very low, low, normal, high, very high, and perfect. Once the worker gives a positive feedback on a recommend information item, the information item will be collected in the "My Favorite" folder. The favorite items are divided into task and documents, which provides a convenient way for the workers to organize the task-relevant knowledge.

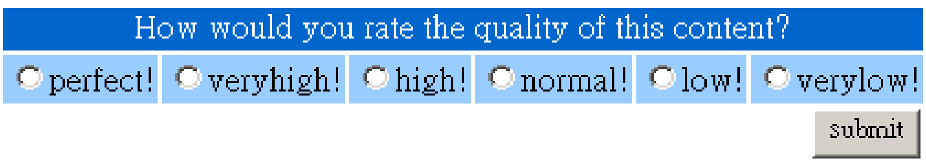

Fig. 6.5 Six Degrees of Relevance Feedback

- **Information Sharing of System**

The information sharing is based on the workers' feedbacks. The system will compute the task peer group based on the analyzing of workers' profile, as introduced in Section 4.4. Fig 6.6 shows the interface of information sharing in task-based workspace. The upper is suggested and expanded information structure which starts from the field level and ends in the task level. The worker can see the viewpoint of others in the same task peer group with associated feedback perceptions. The lower tree in the left frame of Fig. 6.6 is the document set shared from task peer group. Knowledge resources shared from other workers in the same peer group are important information to help the execution of tasks.

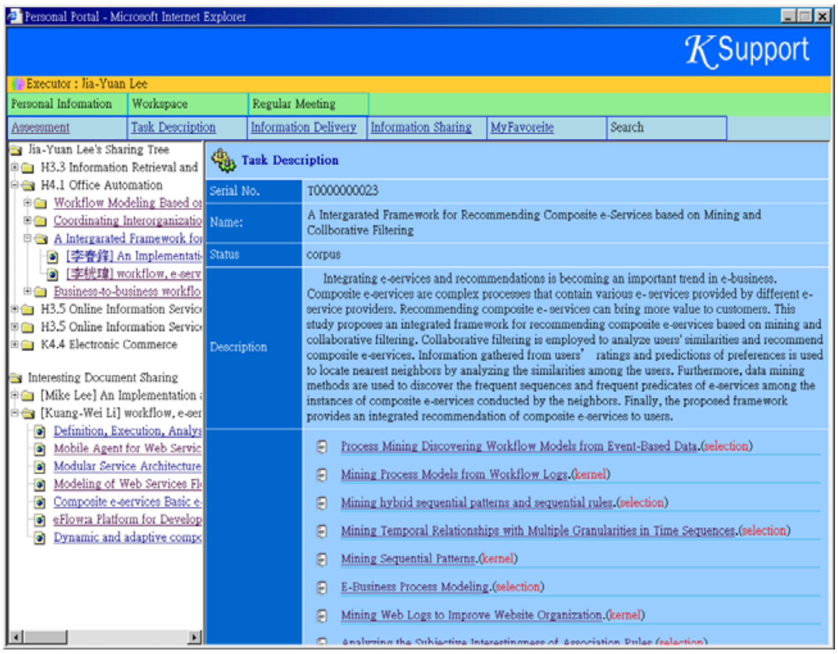

Fig. 6.6 Interface of Information Sharing in Task-based Workspace

7. Conclusion and Future Work

In this work, we present a task-based K-Support system to acquire, model and disseminate codified knowledge among workers during task performance. The proposed K-Support system is designed based on a general architecture that can be tailored to manage different codified knowledge for knowledge support. We manage the tacit and explicit knowledge resources, and implement a profile mechanism and an information structure to achieve knowledge sharing and organizing. With the task-based K-Support portal, workers can access and organize their own task-relevant knowledge as well as share their knowledge among task peer groups. We hope the developed system provides effective knowledge services for facilitating knowledge reuse and sharing in task-based environments.

In order to match the needs of large organization, we can deploy the technology of distributed system. Local workers can not only find the knowledge items in the local host, but also receive the information from remote departments. The peer groups can also be divided by regions. Some access control criteria can be applied on such distributed system to manage the sharing among different branches.

Besides, it is important to resolve the problem of knowledge reformation when modern knowledge items are imported into repository. A re-organization method is needed to integrate the old and new knowledge items. The heterogeneous repositories must be integrated to provide a uniform information structure.

Some future works along with this research should be continued, including:

- Examining the impact of the proposed system in support of knowledge management process.
- Extending the proposed approach to support context-aware or process-aware delivery of task-relevant knowledge.

References

[1] A. Abecker, H. Bernardi, M. Maus, C. Sintek, Wenzel, Information Supply for Business Processes: Coupling Workflow with Document Analysis and Information Retrieval, *Knowledge Based Systems* 13 (5) (2000) 271-284.

[2] S. Agostini, G.. Albolino, De Michelis, De F.R. Paoli, Dondi, Knowledge Management: Stimulating Knowledge Discovery and Sharing, *Proceedings of the International ACM SIGGROUP Conference on Supporting Group Work* Sanibel Island Florida USA November 09-12 (2003) 248–257.

[3] R. Baeza-Yates, B. Ribeiro-Neto, *Modern Information Retrieval*, Addison-Wesley, 1999.

[4] J.M. Bloodgood, W.D. Salisbury, Understanding the Influence of Organizational Change Strategies on Information Technology and Knowledge Management Strategies, *Decision Support Systems* 31 (1) (2001) 55–69, May.

[5] M.J. Benner, and M.L. Tushman, (2000), Exploitation, Exploration, and Process Management: The Productivity Dilemma Revisited, *Academy of Management Review* (in press). Wharton School Working Paper.

[6] T.H. Davenport, D.W. De Long, and M.C. Beers, (1997) Building Successful Knowledge Management Projects, *Sloan Management Review*, pp.43-57.

[7] T. H. Davenport, and L. Prusak, (1998), *Working knowledge: How Organizations Manages What They Know*. Boston MA: Harvard Business School Press.

[8] C.A. Ellis, S.J. Gibbs and G.L. Rein, (1991), Groupware: Some Issues and Experiences, *Communications of the ACM*, 34 (1), pp. 38-58.

[9] Kurt D. Fenstermacher and C. Marlow. (1999), Supporting Consultants with Task-Specific Information Retrieval, *Proceedings of The American Association of Artificial Intelligence*, Orlando Florida: AAAI Press.

[10] G. Fischer, J. Ostwald, Knowledge Management: Problems, Promises, Realities, and Challenges, *IEEE Intelligent Systems* 16 (1) (2001) 60–73.

[11] Gartner Group (Summer 1999), *Knowledge Management Reports*.

[12] D. Goldberg, B.M. Nichols, Oki, and D. Terry, (1992), Using Collaborative Filtering to

Weave an Information Tapestry, *Communications of the ACM*, 35(12), pp.61–70.

[13] P. H. Gray, (2001), The Impact of Knowledge Repositories on Power and Control in the Workplace, *Information Technology & People*, 14(4), pp.368–384.

[14] J. Hahn, and M. Subramani (2000), A Framework of Knowledge Management Systems: Issues and Challenges for Theory and Practice, *Proceedings of the 21st International Conference on Information Systems*, Brisbane, Australia, pp.302–312.

[15] U. Hanani, B. Shapira, and P. Shoval, (2001), Information Filtering: Overview of Issues, Research and Systems, *User Modeling and User-Adapted Interaction (UMUAI)*, 11(3), *Decision Support Systems*, pp.203–259.

[16] K. Hoashi, K. Matsumoto, N. Inoue, and K. Hashimoto, (2000), Document Filtering Method Using Non-Relevant Information Profile, *Proceedings of the 23 rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp.176–183.

[17] D.-R., Liu, I.-C, Wu & K.-S, Yang, The K-Support System: Stimulating Task-based Knowledge Retrieval and Sharing based on Ontological Profiling, 2004, *Institute of Information Management, NCTU, Working Paper*.

[18] J.G. March, (1991), Exploration and Exploitation in Organizational Learning, *Organization Science*, 2(1), pp. 71-87.

[19] I. Nonaka, (1994), A Dynamic Theory of Organizational Knowledge Creation, *Organization Science*, 5(1), pp.14–37.

[20] M. Pazzani, J. Muramatsu; and D. Billus, (1996), Syskill & Webert: Indentifying Interesting Web Sites, *Proceedings of the Thirteen Nacional Conference on Artificial Intelligence*, Portland, Oregon: AAAI Press., pp.54–61.

[21] M. Polanyi, (1966), *The Tacit Dimension*, Routledge & Kegan Paul Ltd. London.

[22] M.F. Porter,:An Algorithm for Suffix Stripping. *Program* 14(3) (1980) 130–137

[23] P. Resnick and H.R. Varian (March 1997), Recommender Systems, *Communications of the ACM*, 40(3), pp56-58.

[24] T. Rodden, (December 1991), A survey of CSCW Systems, *Interacting with Computers*, 3(3), pp.319-353.

[25] M. Ruffolo, and S. Iiritano, (September 2001), Managing the Knowledge Contained in Electronic Documents: a Clustering Method for Text Mining, *Proceedings of Second International DEXA Workshop on Theory and Applications of Knowledge Management (TAKMA 2001)*, Munich Germany, pp.3-7.

[26] Schafer, J. A. Konstan, J. Riedl, (2000), E-Commerce Recommendation Application, *Journal of Data Mining and Knowledge Discovery*, 5(1/2), pp.115–152,.

[27] F. Sebastiani, (1999), Machine Learning in Automated Text Categorization: A Survey, *Technical report*, Istituto di Elaborazione dell'Informazione, C.N.R., Pisa, Italy.

[28] Shapira, P. Shoval and U. Hanani, (1999), Experimentation with an Information Filtering System that Combines Cognitive and Sociological Filtering Integrated with User Stereotypes, *Decision Support Systems*, 27, pp.5–24.

[29] Salton, G.: *The SMART Retrieval System* PrenticeHall Englewood Cliffs NJ (1971)

[30] G. Salton, C. Buckley, Term weighting Approaches in automatic Text Retrieval. *Information Processing & Management*, 24 (5) (1988) 513–523.

[31] H. Widyantoro, T. R. Ioerger, and J. Yen, (2001), Learning User Interest Dynamics with a Three-Descriptor Representation, *Journal of the American Society for Information Science*, 52(3), pp.212-225.

[32] Wiig, (1993), *Knowledge Management Foundation.* Schema Press.

[33] I.-C, Wu,. D.-R., Liu, A Collaborative Relevance Feedback Approach to Task-driven Recommendation. *Proceedings of 7th Pacific Asia Conference on Information Systems (PACIS'03)* Adelaide South Australia 10-13 July (2003) 52–67

[34] M.H. Zack, Managing Codified Knowledge. *Sloan Management Review* 40(4) (Summer 1999) 45–58.