國立交通大學

資訊管理研究所

博士論文

建立知識流程觀模式協助群體知識支援
Establishing Knowledge-Flow View Model for Collaborative Knowledge Support

研究生： 林志偉

指導教授： 劉敦仁 博士

中華民國一〇一年六月

國立交通大學

資訊管理研究所

博士論文

建立知識流程觀模式協助群體知識支援

Establishing Knowledge-Flow View Model for Collaborative Knowledge Support

研究生： 林志偉
研究指導委員會： 李瑞庭 博士
許秉瑜 博士
陳安斌 博士
楊千 博士
指導教授： 劉敦仁 博士

中華民國一〇一年六月

建立知識流程觀模式協助群體知識支援

Establishing Knowledge-Flow View Model for Collaborative Knowledge Support

研究生　：林志偉　　　　Student：Chih-Wei Lin

指導教授：劉敦仁　　　　Advisor：Dr. Duen-Ren Liu

國立交通大學
資訊管理研究所
博士論文

A Dissertation
Submitted to Institute of Information Management
College of Management
National Chiao Tung University
in Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy
in
Information Management
June 2012
Hsinchu, Taiwan, Republic of China

中華民國一〇一年六月

建立知識流程觀模式協助群體知識支援

研究生：林志偉　　　　指導教授：劉敦仁 博士

國立交通大學資訊管理研究所

摘要

在知識密集的工作環境中，有效地提供工作者所需的知識文件，以協助其工作執行，是知識管理領域中的重要議題。從知識需求的角度分析，知識流代表個別與群體知識工作者在執行工作時，其知識需求與知識參考行為的脈絡。組織運用知識流，可有系統的將工作者的知識需求作精確的表示，亦可有效地藉此運作組織的知識支援體系。然而，在群體合作的環境下，不同工作者依其任務特性或扮演角色的不同，常有不同的知識需求。目前已知的知識流研究，大多只提供單一知識流讓工作者參考，並未考量知識流在團隊合作中的適用性。

本研究提出『知識流程觀』模式，以有效改善知識流研究之不足。此一知識流程觀模式，以知識流為基礎，將工作特性及個別角色納入考量，使不同的工作者對同一知識流可有不同的虛擬知識流來滿足其知識需求。

首先，以知識本體論作為知識流中知識節點抽象化的基礎，來建構基礎知識流，從而系統性的表達工作者的知識需求。

在基礎知識流之上，本研究建構知識流程觀模式並進行理論探討。知識流程觀主要是將基礎知識流中的部分知識節點，依照工作特性的知識需求，進行知識概念的歸納抽象化，以產生虛擬知識節點，並進而產生符合工作者知識需求的虛擬知識流。

為了探討工作者在不同角色時的知識需求，本研究亦提出，『以角色為基礎的知識流程觀』模式，利用角色與知識節點的相關度來產生虛擬知識節點，及分析角色所需知識概念層級與工作應有知識概念層級來推算角色知識需求。

知識流程觀與虛擬知識流是一個創新概念與理論模式，不但可擴展知識流的研究理論，對於組織的知識管理，特別是合作型知識支援的推展具有創新與實務的貢獻。

關鍵詞：知識流、知識流程觀、虛擬知識流程、合作型知識支援、知識管理

Establishing Knowledge-Flow View Model for Collaborative Knowledge Support

Student: Chih-Wei Lin Advisor: Dr. Duen-Ren Liu

Institute of Information Management
National Chiao Tung University

Abstract

In knowledge-intensive working environments, workers need task-relevant knowledge and documents to support their task performance. Thus, how to effectively fulfill workers' knowledge-needs is an important issue in realizing knowledge management in organizations. From a knowledge-needs perspective, a knowledge flow (KF) represents a flow of individual's or group members' knowledge-needs and referencing behavior of codified knowledge in conducting tasks. The flow has been utilized to facilitate organizational knowledge support by illustrating workers' knowledge-needs systematically and precisely. However, conventional knowledge-flow models cannot work well in cooperative teams, which team members usually have diverse knowledge-needs in terms of task functions and roles. The reason is that those conventional models only provide one single view to all participants and do not reflect individual knowledge-needs in teams.

Hence, the novel concepts and theoretical model of knowledge flow view (KFV) are proposed in this dissertation. The KFV model builds virtual knowledge flows derived from a base KF to provide abstracted knowledge to serve different workers' knowledge-needs from task function and role perspectives.

This dissertation uses domain ontology as the base of knowledge node abstraction. Hence, base knowledge flows are built to represent workers' knowledge-needs systematically. Based on the base knowledge flows, a theoretical model of KFV is investigated and developed for discovering virtual knowledge nodes and virtual knowledge flows. The KFV model abstracts the knowledge nodes of partial base knowledge flow to generate virtual knowledge nodes according to task functions, through knowledge concept induction and generalization.

In addition, this dissertation proposes a role-based KFV model to investigate different knowledge-needs of distinct roles. The model exploits the relevance degrees between roles and knowledge nodes to derive virtual knowledge nodes and analyzes roles’ required knowledge concept level and operation required knowledge concept level to derive knowledge concepts of virtual knowledge nodes.

The models of KFV and the concept of virtual knowledge flow are innovative, which extends the scope of knowledge flow research and enhances the efficiency of cooperative knowledge support in organizations.

Keyword: knowledge flow, knowledge-flow view, virtual knowledge flow, cooperative knowledge support, knowledge management

To my lovely family:

Dad & Mom

June, my dear wife

Steven and Selina

Acknowledgement

This dissertation would not have been possible without the guidance and the support of several individuals who in one way or another contributed and extended their valuable assistance in the preparation and completion of this study.

First of all, I would like to express my sincere appreciation to my advisor, Dr. Duen-Ren Liu, for his generous encouragement and constructive inspiration to my PhD study. Special thanks are also given to Professor Anthony J.T. Lee, Professor Ping-Yu Hsu, Professor An-Pin Chen, and Professor Chyan Yang for providing comments to improve my dissertation and serving on my dissertation committee.

Members of Database and Information System Lab also deserve my sincerest thanks, their friendship and assistance has meant more to me than I could ever express.

Special thanks go to my managers and colleagues in tsmc: Ted Sun, T. W. Kuo, Joseph Liao, Roger Lin, Ken Chen, Frank Sung and R. M. Wang for their understanding and inspiration when I was a part-time PhD student.

Last, but not the least, I wish to extend my special and sincere thanks to my parents, my wife, June and my children, Steven and Selina, for their love during my study.

Jacky C. W. Lin

2012. 6. 30

Table of Contents

1896

List of Figures

List of Definitions

Chapter 1 Introduction

1.1 Motivation

In a knowledge-based organization, knowledge workers need to acquire a variety of knowledge (information) about their tasks [14]. Therefore, many organizations have built knowledge support platforms to assist workers in meeting their knowledge-needs. These platforms help workers to identify and share knowledge in order to speed up organization innovation and improve employee productivity [11, 25]. Studies on formulating knowledge-needs and streamlining knowledge provision are becoming more prevalent as the value of knowledge support keeps increasing [2, 38, 43, 49, 62-63].

The fast pace of technology evolution and the short cycle time for solving problems in current knowledge intensive environments has led to an emphasis on teamwork [16, 22]. For example, R&D activities often consist of many knowledge-intensive tasks that must be completed within a limited time period. These tasks are usually conducted through cross-function collaboration. By integrating the expertise and perspectives of various individuals, teams can quickly respond to interdisciplinary problems and enhance decision quality, thus providing a holistic solution. However, due to their individual task functions and roles, many team members have different knowledge-needs; as a result, they may expend considerable effort in seeking and synthesizing knowledge to obtain the required task-relevant knowledge [47, 65]. Reducing this expenditure of effort is one of the main challenges of collaborative knowledge support.

By mapping knowledge flows, organizations can provide task-relevant knowledge to workers that help them fulfill their knowledge-needs quickly and effectively [28]. A knowledge flow (KF) represents the flow of an individual’s or group members’ knowledge-needs and the referencing sequence of codified knowledge in conducting organizational tasks. Knowledge flows are an emerging topic of investigation in the knowledge management research field, and several studies have built knowledge flow models to illustrate knowledge sharing among knowledge workers [25, 29, 33, 40, 42, 69-72]. For example, researchers in scientific fields who propose new ideas through content publishing form a knowledge flow in science [71]. The known ideas in one paper inspire new ideas for other researchers, and the established relationships or links generate a citation chain. Some studies have addressed knowledge sharing by defining the process in which knowledge is transferred from one team member to another [69-70, 72]. Other researchers have focused on discovering knowledge flows by analyzing workers' knowledge-needs; the results have contributed to knowledge sharing in which the codified knowledge becomes available for recommendations to workers [28]. The shortcoming of these studies, however, is that the conventional models provide the same knowledge support to all team members; in other words, they do not consider the individual knowledge-needs that arise in a collaborative environment.

This dissertation proposes a novel knowledge-flow view (KFV) concept to consider workers' knowledge-needs from different aspects, and demonstrates the benefits a cooperative team can receive while adopting them. The novel KFV models not only re-innovate conventional knowledge flow models but also enhance the efficiency of knowledge flow usage, as well as the effectiveness of knowledge sharing and knowledge support in teamwork environments.

1.2 Goals

Driven by the motivation, the dissertation aims to develop the models of knowledge-flow view to facilitate collaborative knowledge support, which fulfills teammates' knowledge-needs from various aspects. Major goals of this work are listed below.

-Theoretically model base knowledge flows by adopting domain ontology to formulate knowledge-needs precisely.

-Develop an essential knowledge-flow view (KFV) model to derive virtual knowledge flows from a base knowledge flow in terms of task functions.

-Put roles in perspective to build role-based knowledge-flow view (r-KFV) model for addressing the relationships among roles, operations and knowledge requirements.

-Supply team participants with knowledge at required granularity to support task performance and team communication.

1.3 Approaches

This work extends previous knowledge flow research by exploring how to enhance conventional knowledge flow models to satisfy workers’ different knowledge-needs in teamwork environments. The challenges in a collaborative team are considerable and they pose many barriers to knowledge flows [31, 48]. Two of these barriers are low effectiveness and poor communication. Team members require different conceptual levels of knowledge to perform tasks and communicate with each other. For example, workers need specific knowledge to perform their tasks and general knowledge to communicate with other workers whose tasks or roles differ from their own. Effectively making collaborative knowledge provision at both specific level and general level is the key to team performance and productivities.

To formulate knowledge-needs precisely and model knowledge flows formally, this work first proposes a base knowledge flow (BKF) model which adopts domain ontology to describe knowledge-needs by the composition of knowledge concepts. According to the BKF model, a knowledge flow designer (KF designer) may either consult domain experts or investigate workers’ document access logs to identify participants’ knowledge-needs. Thus, the collection of knowledge-needs and the order of referencing sequences would be used to construct base knowledge flows, which represent the knowledge-needs of participants by knowledge concepts.

In addition, since the BKF model does not consider personalized requirements and provides only one single view of a base knowledge flow, this would impact the effectiveness of knowledge provision while applying it in collaborative environments. Therefore, by considering the different conceptual levels of knowledge in illustrating individual knowledge-needs, this work establishes an essential knowledge-flow view (KFV) model that aims to generalize knowledge concepts and derive virtual knowledge flows mainly from a task function perspective; as such, the essential KFV model would be capable of serving individuals' knowledge-needs. A similar concept exists in database management systems, where administrators generate virtual database views from a base table to serve different purposes. A virtual knowledge flow (virtual KF) is derived dynamically from a base knowledge flow (base KF) according to the essential model which is employed to abstract knowledge concepts. The novel essential KFV model uses an order-preserving approach and a knowledge concept generalization mechanism to abstract some base knowledge nodes in a base KF, thus generating virtual knowledge nodes that correspond to the individual knowledge-needs of different workers [34].

Last, in real practice, tasks are often assigned to dedicated roles to ensure quality and security. Therefore, if a task involves teamwork, workers' knowledge-needs will vary, depending on the roles they play [30]. For example, in a computer manufacturing company, a role of engineering is responsible for product development and another role of marketing designs strategies to launch and promotes new products. In this scenario, the engineering role needs a specific level of technical knowledge, but the marketing role only needs a general level of such technical knowledge to communicate with engineers. Thus, a role-based knowledge-flow view (r-KFV) model is required in the context, which includes the aspect of roles to apply knowledge management applications in teams [27]. The r-KFV model analyzes the conceptual levels of knowledge required by workers based on their roles, and develops role-based knowledge flow abstraction methods that generate virtual knowledge nodes to provide the appropriate level of knowledge for each role.

To investigate the feasibility of the proposed BKF and KFV models, a preliminary analysis was conducted. A case of mobile phone development and system design-related documents were illustrated and provided to several professionals to ask for their opinions about the feasibility of the proposed models from a practical perspective. Overall, there was general agreement with the feasibility of the KFV models. The agreement, to some extent, validates the feasibility of the approaches.

In summary, this work addresses an important extension of knowledge flow research. It considers a phenomenon that workers in teams usually have different knowledge-needs for task execution and team communication in terms of roles and task functions. The concept of knowledge-flow view and related models are proposed to support such knowledge-needs in teamwork environments.

1.4 Contributions

For a new discipline or a new research topic, theoretical papers are required to explore the basic theory by illustrating term definitions and establishing relationships between concepts [13, 19]. Thus, in order to explore the new topic – knowledge-flow view, this work is targeted as a theoretical research to establish BKF model, essential KFV model and r-KFV model for extending knowledge flow research in cooperative teams for organizational knowledge support.

This dissertation contributes to knowledge management development, first by showing how a knowledge flow can address knowledge-needs. The previous studies are lacking in illustrating knowledge flows in terms of workers' knowledge-needs. The proposed BKF model fills this gap and helps researchers to obtain a clear view of knowledge flow research.

Additionally, this study investigates the shortage of knowledge support in collaborative teams because the workers in a team usually have different knowledge-needs according to their task functions. The essential KFV model is proposed to address the shortage. According to the essential KFV model, KF designers generate virtual knowledge flows that conceal confidential or detailed information base on workers' task functions. Through an order-preserving approach and a knowledge concept generalization mechanism, the virtual knowledge flows not only comply with organizational information security policy but also reflect the granularity of knowledge-needs. Thus, the essential KFV model can advance the applicability of knowledge flow research to cooperative knowledge support environments.

The r-KFV model examines workers' knowledge-needs in terms of their roles. The role represents a duty or a job position with the authority and responsibility to perform certain jobs within a team. So, it is essential to conduct knowledge provision in teams from a roles perspective. The r-KFV model design a kernel approach to derive role-based virtual knowledge flows from a base knowledge flow in role and operation perspectives. Based on role-operation knowledge requirement, the r-KFV model accurately illustrates roles' knowledge-needs and effectively facilitates knowledge concept abstraction. It is an originative study of roles to address an important extension of knowledge flow research.

This work facilitates collaboration in teams by effective knowledge support. The innovative concept of knowledge-flow view and the proposed theoretical models can enhance the scope of knowledge flow research. In addition, this work also improves the efficiency of knowledge flowing, as well as the effectiveness of knowledge sharing and knowledge support in organizations.

1.5 Organization

Figure 1 shows the research framework including literature review in *Related work* part, model development and methodology design in *Modeling* part, and model evaluation in *Preliminary analysis* part.

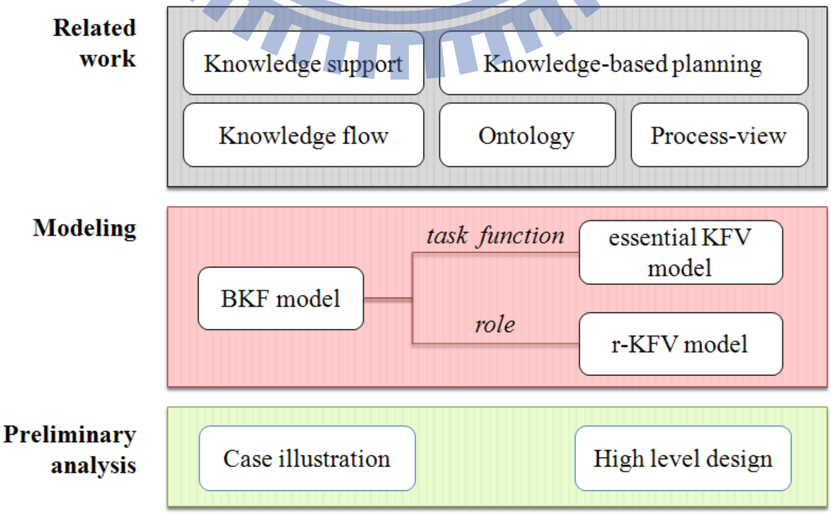


Figure 1. Research framework.

The rest of this dissertation is organized as follows: Chapter 2 contains a review of related work. Chapter 3 builds a formal base knowledge flow (BKF) model. Chapter 4 defines and analyzes an essential knowledge-flow view (KFV) model. The algorithms to generalize knowledge concepts and derive virtual knowledge flows are described. Chapter 5 discusses the concepts of role-based knowledge-flow view (r-KFV) model and the methods of generating role-based virtual knowledge flows. Conclusions and future work are made in Chapter 6.

Chapter 2 Related work

This chapter provides a brief summary of related research: knowledge management and knowledge support, knowledge flow, knowledge-based planning, ontology, process and process-view.

2.1 Knowledge management and knowledge support

Knowledge is one of the key assets to ensure sustained competitive advantage in the highly technological and global environment of modern organizations [21, 32, 50, 59]. To achieve success in this environment, workers need to effectively apply knowledge to conduct knowledge-intensive operations and management activities [9, 38, 70].

Knowledge management (KM) supplies the principles of creation, organization, transfer and application of the knowledge within organizations [26] and is recognized as a crucial practice for enabling organizations to survive in a knowledge economy era [64]. One purpose of KM is to support workers in fulfilling their knowledge-needs, by bridging the gap between workers' knowledge and the requirements of tasks [2, 58, 63]. Studies have shown that precise and timely knowledge support is an important mechanism for increasing both productivity and work effectiveness [28, 38].

In a task-based business environment, tasks are conducted in work processes. The effective provision of task-relevant knowledge and context information is crucial to increasing workers' productivity. To meet this provision, integration solutions of information retrieval (IR) and workflow management systems (WfMS) have been developed. These solutions proactively deliver task-relevant knowledge according to the context of tasks [1, 43]. For example, the KnowMore system derives task profiles from process definitions that facilitate knowledge provision [1]. The Flow-Wiki system was developed by a wiki-based approach for agilely managing workflows and effectively providing relevant information to participators [24]. In this way, process participants can obtain knowledge that pertains to task profiles and/or the execution context of the current process.

Liu et al. [62-63] proposed a task-based *K*-support system that provides knowledge to adaptively meet a worker's dynamic information needs by analyzing his/her access behavior and relevance feedback on documents. Furthermore, because of the nature of teamwork, a collaborative mechanism is essential for establishing knowledge management systems [4, 67].

2.2 Knowledge flow

Knowledge flow research focuses on how knowledge flows transmit, share and accumulate knowledge in a team. In a workflow situation, working knowledge may flow among workers, while process knowledge may flow among various tasks [70, 72-73]. Thus, the knowledge flow reflects the level of knowledge cooperation between workers or processes, and influences the effectiveness of teamwork or workflow.

To fulfill workers' knowledge-needs, knowledge flows provide links among knowledge sources. Through knowledge flows, workers can effectively obtain knowledge from these sources to execute tasks [25]. Knowledge flows illustrate the sequence of knowledge-needs and/or the order of referring documents when workers perform tasks. Knowledge flows can facilitate knowledge sharing and reuse in both business and research environments. For example, Zhuge [70] illustrated a knowledge flow within a software development team of a distributed organization. Here, the knowledge flow carried and gathered knowledge from one team member to another for sequential knowledge sharing. Similar knowledge sharing can take place in a citation chain where knowledge is transferred among scientific researches. In this context, the citation chain of papers is a knowledge flow that disseminates knowledge among scientists and inspires new ideas [71].

Several knowledge flow models have been built in recent researches. Luo et al. [40] modeled a Textual Knowledge Flow (TKF) from a semantic link network. The purpose of the TKF was to recommend proper browsing paths to users after evaluating their interests and inputs. Lai and Liu [28] constructed a time-ordering knowledge flow model to illustrate the sequence of workers' knowledge referencing behaviors. In this model, workers obtained proper knowledge to fulfill their knowledge-needs through the knowledge flows discovered in document access logs. Kim et al. [25] proposed a knowledge flow model using a process-oriented approach to capture, store and transfer knowledge. Zhang et al. [66] used Petri-Net to model a knowledge flow. In this model, a knowledge node was used to generate, learn, process, understand, synthesize and deliver knowledge based on four types of flow relations: creation, merging, replication and broadcasting. Zhao and Dai [68] integrated business processes and knowledge flows and divided knowledge flows into sequence, distribution, combination and self-reflection patterns based on RAD (role-activity-diagram) model. Finally, Anjewierden et al. [5] suggested that the referencing sequence in weblogs may be regarded as a knowledge flow and can be described as a sender-message-receiver model.

2.3 Knowledge-based planning

Both knowledge flow and knowledge-based planning prompt similar ideas about embedding knowledge while building models. Knowledge-based planning is a planning methodology used to identify a sequence of tasks executed by one or more agents under given initial conditions and resource constrains to achieve final goals [6]. The methodology involves knowledge acquisition, knowledge validation and knowledge maintenance of planning domains, and adopts appropriate knowledge-based planning tools to build planning models [6]. For example, R-Moreno et al. [46] successfully utilized a planning and scheduling system as well as a workflow modeling tool to plan a telephone installation workflow model. The workflow modeling tool was used to acquire relevant knowledge, such as initial conditions, resource constrains and final goals; then the planning and scheduling system was used to convert the knowledge into planning standard expressions. A knowledge-based planning system can also be employed to manage the result of planned tasks for the purpose of fulfilling other tasks’ preconditions. Chow et al. [12], for example, proposed a strategic knowledge-based planning system (SKPS) that combined knowledge rules with mathematical models to formulate co-loading shipment plans. Through SKPS, shipment planners could acquire, validate and maintain knowledge of the shipment domain, and thus build a co-loading shipment planning model so that executors could utilize the knowledge in the model to perform tasks efficiently.

As the above examples demonstrate, knowledge-based planning focuses on building planning models for problem solving or task execution. Knowledge flow research contributes to the building of knowledge flow models for corresponding task execution plans (or workflow processes) that support knowledge provision, sharing and transferring [28, 70]. Knowledge flows can be either derived by mining workers' access logs [28] or specified by KF designers according to their experience in executing the corresponding workflow process [69, 72]. Besides these two methods of deriving knowledge flows, knowledge-based planning tools can complement knowledge flow research by helping designers build the appropriate knowledge flows that correspond to task execution plans.

2.4 Ontology

Ontology is a widely accepted approach for capturing and representing knowledge possessed by an organization [44, 54]. It is a conceptualization mechanism that defines knowledge concepts in a specific domain and constructs a hierarchical structure to describe their inter-relationships [18]. Ontology can promote a common understanding throughout a whole organization to facilitate knowledge storage, retrieval and synthesis [45]. For example, the common terminologies and knowledge concepts in ontology can improve the problem-solving capability and efficiency within a supply chain [7]. Another example of ontology pertains to the knowledge concepts derived from Wikipedia articles and categories, which can be used to predict the contents of documents [55].Weng and Chang [60] proposed a research document recommendation system which exploited ontology to construct user profiles, and utilized the profiles to illustrate researchers' interests. Afacan and Demirkan [3] developed an ontology-based universal design support system to support designers in the conceptual design phase; it adopts ontologies to process and represent required knowledge. As the above examples illustrate, ontology is a versatile paradigm that can be applied in many domains.

Building ontology is an evolving process and involves many techniques and tools to facilitate the whole process. Obviously, the construction process would include an evaluation and feedback mechanism to gradually improve ontology quality and obtain common understanding in organizations [29, 45, 56]. For example, Uschold and King [57] proposed a skeletal methodology to build an enterprise ontology; it comprises four phases: scoping, building, evaluating and documenting. Du et al. [15] designed a six-phase process that includes the preparation, transformation, clustering, recognition, refinement and revision for extracting ontology from unstructured HTML pages. Therefore, involving users in the evaluation or refinement phase is essential for gradually adjusting the quality of ontology. Many ontology-building tools, such as Protégé, OntoEdit and SNet-Builder, can effectively support the ontology construction process to serve predefined purposes and meet users' requirements [10, 44].

2.5 Process and process-view

Recently, business process modeling has been rapidly applied to streamline business administration and to facilitate cooperation among enterprises. Business process modeling refers to the design, analysis and execution of business processes [20]. Its goals are to describe a set of activities that can be performed in sequence, and to allocate resources and arrange jobs optimally by analyzing the organizational and technical environments [61]. By employing appropriate modeling tools, business process modeling can provide pre-defined templates that allow enterprises to enact their business processes in an effective and efficient manner.

In an industrial environment, processes describe the flows of business operations. Workflow management systems are definition and execution tools that support these operations [45]. In practice, participants involved in a workflow need a flexible workflow model capable of providing appropriate process information [2, 36]. Because of the increasing complexity of business processes and the variety of participants, it is beneficial for organizations to define virtual processes with different views of the workflow [8, 17, 36, 52]. Liu and Shen [36] presented a novel concept of process abstraction: the process-view.

A process-view is an abstracted process derived from a base process to provide generalized process information. The process-view is generated by an order-preserving approach, which ensures that the original order of the activities in the base process is preserved. Under the process-view concept, a WfMS can provide various views of a process for different participants within an organization or cross organizations [37]. Shen and Liu [53] proposed a role-based approach to discover role-relevant process views for different workflow participants. The role-based approach generates process view automatically, based on the relevance degrees between roles and tasks. This work adopts similar ideas to generate virtual knowledge flows from a base knowledge flow, while retaining the knowledge referencing order.

Chapter 3 Base knowledge flow model

In cooperative working environments, a base knowledge flow (base KF) represents the flow of team members' knowledge-needs and the referencing sequence of codified knowledge that workers need while conducting business processes or research tasks. To formulate knowledge-needs precisely and model knowledge flows formally, this chapter illustrates a base knowledge flow (BKF) model which adopts domain ontology to describe knowledge-needs by a composition of knowledge concepts.

3.1 A motivation example of base knowledge flow

A mobile phone development process consists of multiple tasks which require joint efforts from Marketing, Design, Outsourcing, Quality Assurance and Sales departments. Participants not only contribute their expertise, but also refer to additional codified knowledge that contributes to the performance of tasks in processes. The flow of knowledge-needs and the sequence of document reference can be represented by a base KF. Figure 2 shows the mobile phone development process consisting of nine tasks: business analysis, industrial design, major parts identification, parts sourcing, hardware design, platform setup, application design, verification and commercialization.

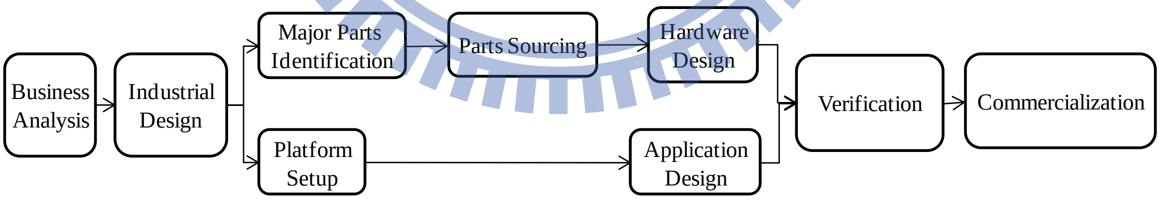


Figure 2. Mobile phone development process.

In the above process, team members may have the knowledge-needs of *marketing segmentation* and *consumer analysis* while conducting the business analysis task. The knowledge concepts relevant to the knowledge-needs include: *geographic segmentation*, *psychographic segmentation*, *consumption environment*, and *consumer behavior*. Knowledge flow designers (KF designers) put these knowledge concepts into a base knowledge node to represent the knowledge-needs of the business analysis task. In addition, the team members may also have the knowledge-needs of accessing two related knowledge concepts: *compliance guidance* and *usability checklist*, while performing the verification task [23]. The knowledge of how to build base KFs is derived from structured interviews and workshops [25], system event logs [28], as well as the content of tasks. For example, in investigating the whole business process, KF designers rely on their experience [69, 72], interviews of domain experts [25], and/or the analyses of workers' document access logs [28, 33] to collect knowledge-needs on a task-by-task basis. These knowledge-needs are illustrated by knowledge concepts which are identified by domain ontology. By using domain ontology, KF designers group relevant knowledge concepts into corresponding base knowledge nodes to form a base KF.

Figure 3 shows the corresponding base KF of the mobile phone development process. In the base KF, for example, the knowledge concept *consumer behavior* is related to market trends research and customer preferences investigation, which facilitate marketing staff and designers in identifying major parts such as display, battery and cards options by evaluating their combinations. Accordingly, KF designers group the relevant knowledge concepts *consumer behavior*, *display options, battery options* and *card options* to form the base knowledge node k_2 to represent the knowledge-needs of conducting the major parts identification task.

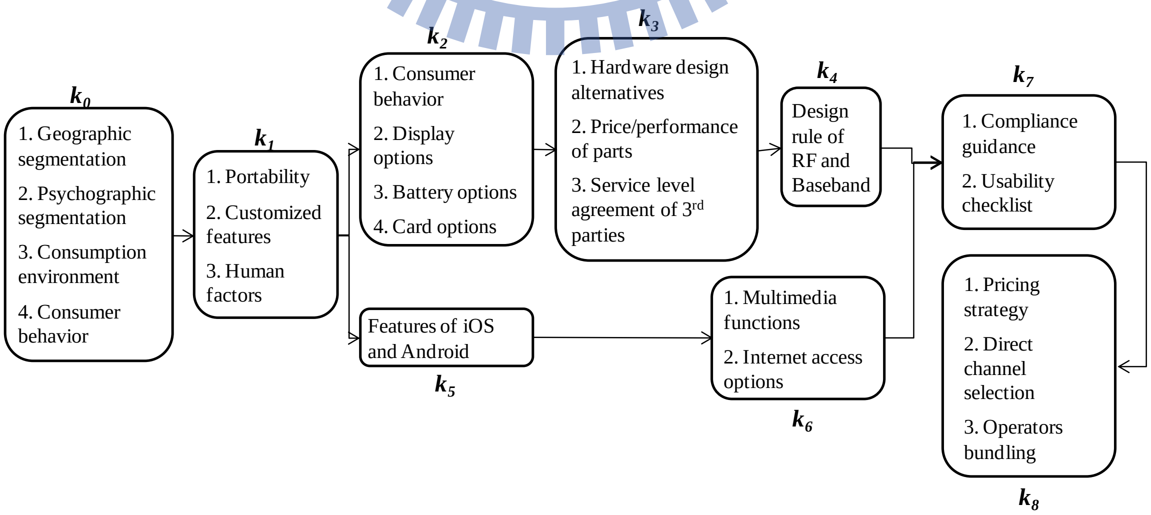


Figure 3. Base KF of the mobile phone development process.

3.2 Define base knowledge flow model

This section formally defines domain ontology and base KF for the purpose of building a theoretical BKF model. Definition 1 models domain ontology, which is the infrastructure for sharing knowledge concepts throughout the whole organization. Definition 2 – Definition 7 formulate the BKF model.

Definition 1: Domain ontology

Ontology is constructed to define knowledge concepts and their hierarchical relationships in a domain.

Ontology is defined as $O = <C, HR>$, where C is a set of knowledge concepts derived from a specific domain. HR is a set of hierarchical relations which define the parent-child relationships among knowledge concepts in C, and HR is formally expressed by $HR = \{hr \mid hr \in C \times C\}$.

For two knowledge concepts, x and y, if x has a downward link to y (or y has an upward link to x) in an ontology, then x is the parent concept of y and y is the child concept of x. Two semantic relations, *Generalization* and *Specialization*, are used to describe the relative conceptual level of two knowledge concepts. Relations between the parent concept x and the child concept y are formally expressed by $Specialization\ (x) = \{y \mid y \text{ is a child concept of } x\}$ and $Generalization(y) = \{x \mid x \text{ is a parent concept of } y\}$.

Figure 4 shows an example of the domain ontology in the mobile phone development domain. The root of the ontology is *mobile phone development*. It represents the most general knowledge concept, as indicated also by R&D strategy and the product development guideline. Six subconcepts, *marketing*, *industrial design*, *hardware design*, *software design*, *quality verification* and *sales* appear under *mobile phone development*. Likewise, *market segmentation*, *consumer analysis* and *outsourcing* are the subconcepts of *marketing*. Hence, *Specialization* (*marketing*) = {*market segmentation*, *consumer analysis*, *outsourcing*} and *Generalization* (*market segmentation*) = {*marketing*}.

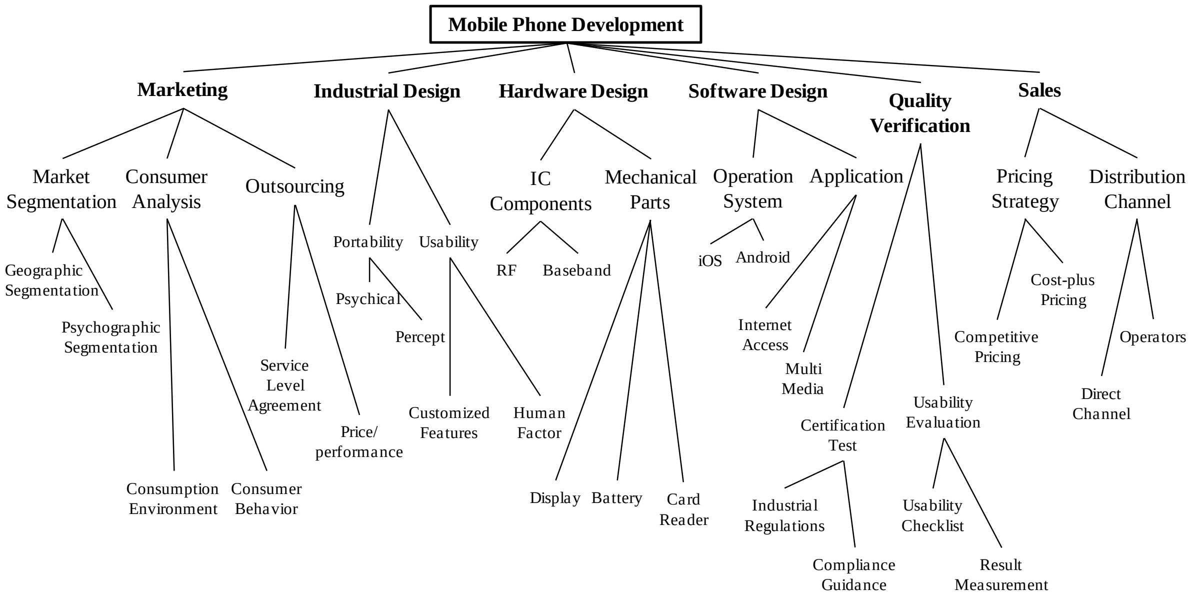


Figure 4. Domain ontology of mobile phone development.

In existing research [29, 39, 41, 51, 54], the relations of ontologies are designated as: *is-a*, *part-of*, *subclass*, *synonym* or *related-to*. The meanings of these relations pertain to the design purpose of ontologies and the characteristics of knowledge concepts. In this work, ontology is designed to represent knowledge concepts and their hierarchical relationships in a domain. The use of ontology facilitates the abstraction of knowledge concepts based on their conceptual levels, which are required for building virtual knowledge flows. Two semantic relations, *Generalization* and *Specialization*, are used to describe the relative conceptual levels of two knowledge concepts without distinguishing between the meanings of relations such as *is-a*, *subclass*, *synonym* or *related-to*. According to the *Generalization* relation, child (specific) knowledge concepts can be abstracted to parent (general) knowledge concepts. For example, the knowledge concept *operation system* contains knowledge about categories and functions of APIs. Knowledge concept *iOS* contains knowledge of detailed specifications of Apple iOS's APIs. Thus, these two knowledge concepts are related and on different conceptual levels. The knowledge concept *operation system* comprises more general knowledge than does knowledge concept *iOS*. Thus, knowledge concept *operation system* is on a higher conceptual level than knowledge concept *iOS*. The same relation exists between knowledge concepts *operation system* and *Android*. As shown in Figure 4, knowledge concepts *iOS* and *Android* have upward links to knowledge concept *operation system*. The relations among them can be expressed as *Generalization* (*iOS*) = {*operation system*} and *Specialization* (*operation system*) = {*iOS*, *Android* …}.

Based on domain ontology, KF designers can formulate workers' knowledge-needs by using combinations of knowledge concepts on different conceptual levels. For example, workers with knowledge-needs about market segmentation might identify and divide potential consumers into groups according to their characteristics, behavior and location. In this example, knowledge-needs can be represented either by the knowledge concept *market segmentation* or by two knowledge concepts, *geographic segmentation* and *psychographic segmentation*. The knowledge concept *market segmentation* is a general concept. As such, it is used to describe the purpose of segmentation or to introduce the guideline of selecting one segmentation alternative among others. By contrast, the knowledge concepts, *geographic segmentation* and *psychographic segmentation*, are specific concepts that describe detailed knowledge pertaining to the steps used in analysis or the segmentation criteria. *Market segmentation* is the parent (general) concept of *geographic segmentation* and *psychographic segmentation*, whereas *geographic segmentation* and *psychographic segmentation* are the child (specific) concepts of *market segmentation*.

As this example shows, workers' knowledge-needs can be expressed as a combination of knowledge concepts in domain ontology, where the conceptual levels of these knowledge concepts can be identified by their positions in the ontology. By grouping knowledge concepts at the proper conceptual levels, KF designers can use domain ontology as a reference base to identify workers' knowledge-needs. Furthermore, domain ontology can facilitate the abstraction of knowledge concepts, which are required for generating virtual knowledge flows.

It is notable that the structure of domain ontology could be a tree as shown in Figure 4 or a lattice as shown in Figure 5. In Figure 5, the concept *H* has two parent concepts *D* and *E*. The semantic relations, *Generalization* and *Specification*, can be applied in the lattice structure: *Generalization* (*H*) = {*D*, *E*} and *Specification* (*D*) = {*G*, *H*}. Hence, domain ontology with either tree structure or lattice structure can be properly used for knowledge concept abstraction. This work adopts tree structure to simplify illustration.

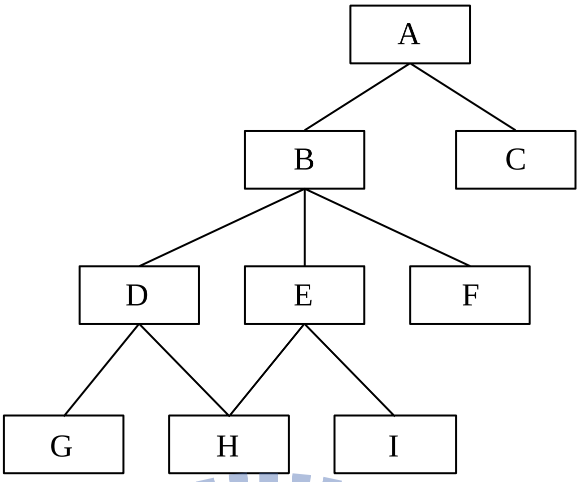


Figure 5. Domain ontology with lattice structure

Next, the BKF model is formulated through a series of definitions provided below:

Definition 2: Base knowledge node

A base knowledge node (base KN) x is a set of knowledge concepts needed by workers to fulfill their tasks. The knowledge concepts of x are denoted as $KC(x) = \{c_1, c_2, c_3, \ldots, c_m\}$ where knowledge concept c_i can be identified by domain ontology.

Definition 3: Dependency

A dependency is an ordered pair (base KN x, base KN y) denoted by $dep(x, y)$. This notation indicates that after knowledge concepts in x have been referenced, workers can start to reference the knowledge concepts in y. In $dep(x, y)$, x is called the preceding node and y is called the succeeding node.

Definition 4: Base knowledge flow

A base knowledge flow (base KF) is a 2-tuples <*KNS*, *DS*>, where *KNS* is a nonempty set, and its members are base KNs in the base KF. *DS* is a nonempty set, and its members are dependencies.

Definition 5: Neighboring

Two base KNs are neighboring if a dependency between them exists in *DS*.

Definition 6: Path

Given a base knowledge flow $BKF=<KNS, DS>$, a path is defined to include a starting base KN k_0, intermediate base KNs $k_1, k_2, \ldots, k_{n-1}$, an ending base KN k_n and a set of dependencies, $dep(k_{i-1}, k_i) \in DS$, for $i = 1,2,\ldots n$. The path from k_0 to k_n is denoted by $k_0 \rightarrow k_n$.

Definition 7: Ordering relation

Given a base knowledge flow $BKF=<KNS, DS>$ and two base KNs $x, y \in KNS$, x has a higher order than y if a path $x \rightarrow y$ exists. The ordering relation is denoted as $x > y$.

3.3 Discussion

In the motivation example, KF designers derive the base KF's knowledge dependencies from the process level dependencies because it is a more intuitive and easier way for team members to understand. Nevertheless, KF designers can apply different ways to set the knowledge dependencies from other perspectives. Generally speaking, the knowledge dependencies in a base KF indicate the referencing sequence of knowledge (information) in task performance, which may occur in a distributed software development team [69], an academic research project [28] or a web exploration [5]. So, knowledge dependencies do not always relate to process level dependencies. In practice, KF designers are responsible for setting knowledge dependencies based on the characteristics of applications and act as consultants to provide KF and facilitate knowledge provision to teams. Actually, project team takes major responsibility to conduct tasks and deliver results.

This chapter contributes to the research of knowledge flow, first by showing how a knowledge flow can address knowledge-needs. In previous literature, models that formally illustrate a knowledge flow and corresponding knowledge-needs of workers together are lacking. The proposed BKF model fills this gap by including three initiatives: (1) it adopts domain ontology to describe knowledge-needs by a composition of knowledge concepts; (2) it derives base KNs from the activities in processes to visually display workers' knowledge-needs; and (3) it defines flow dependencies as the sequence of an individual's or group members' knowledge-needs and/or the order of referencing codified documents. The flow dependencies, to some degree, can help workers set the priority of information accessing and get latest information. The BKF model can help organizations assess their current practices of knowledge sharing and knowledge reuse to gain insights into the required knowledge concepts and build appropriate knowledge flows. It also paves the way for researchers to obtain a clear view of knowledge flow research.

Chapter 4 Knowledge-flow view model

Views in database management system are *virtual tables* generated from either *base tables* or previously defined views to serve different purposes. Similarly, views of knowledge flow are derived from either *base knowledge flows* or other knowledge-flow views, and are considered *virtual knowledge flows*. That is, a virtual knowledge flow (virtual KF) is an abstracted knowledge flow generated from a base knowledge flow (base KF), and is used to reveal abstracted knowledge. The base KFs and the base knowledge flow (BKF) model have been introduced in previous chapter. Furthermore, this chapter presents a knowledge-flow view (KFV) model to build virtual KFs by abstracting base knowledge nodes (base KNs) in a base KF. The KFV model generates corresponding virtual knowledge nodes (virtual KNs) through an order-preserving approach and a knowledge concept generalization mechanism. The virtual KFs not only fulfill workers' different knowledge-needs but also facilitate knowledge support in teamwork.

4.1 Virtual knowledge flow: abstracted form of base knowledge flow

By knowing what other members know, a team is able to gain better decision quality and communicate more effectively [47]. Therefore, team members not only need specific knowledge to conduct their tasks, but also require general knowledge about tasks performed by other members to facilitate their communication. For example, in the mobile phone development process, marketing staff members refer to specific geographic segmentation documents to identify possible consumer groups, and gather specific knowledge of consumer behavior to determine the acceptance level of a new mobile phone. In addition, they need general knowledge related to *industrial design*, *hardware design*, *software design*, *quality verification* and *sales* to communicate with members outside their departments through the use of common terminology. The knowledge support of both specific and general knowledge pertaining to different tasks can assist marketing staff members to complete their business analysis task and increase the communication quality of the team. However, since conventional knowledge flow models provide only a single view of a knowledge flow and do not consider personalized requirements, they are not applicable in such environments. In fact, project managers do not need specific and detailed knowledge about business analysis, industrial design, hardware design and other tasks. They only need general knowledge of these tasks to help them make decisions and communicate with other team members. Figure 6 shows a virtual KF with general knowledge concepts that can meet project managers' knowledge-needs.

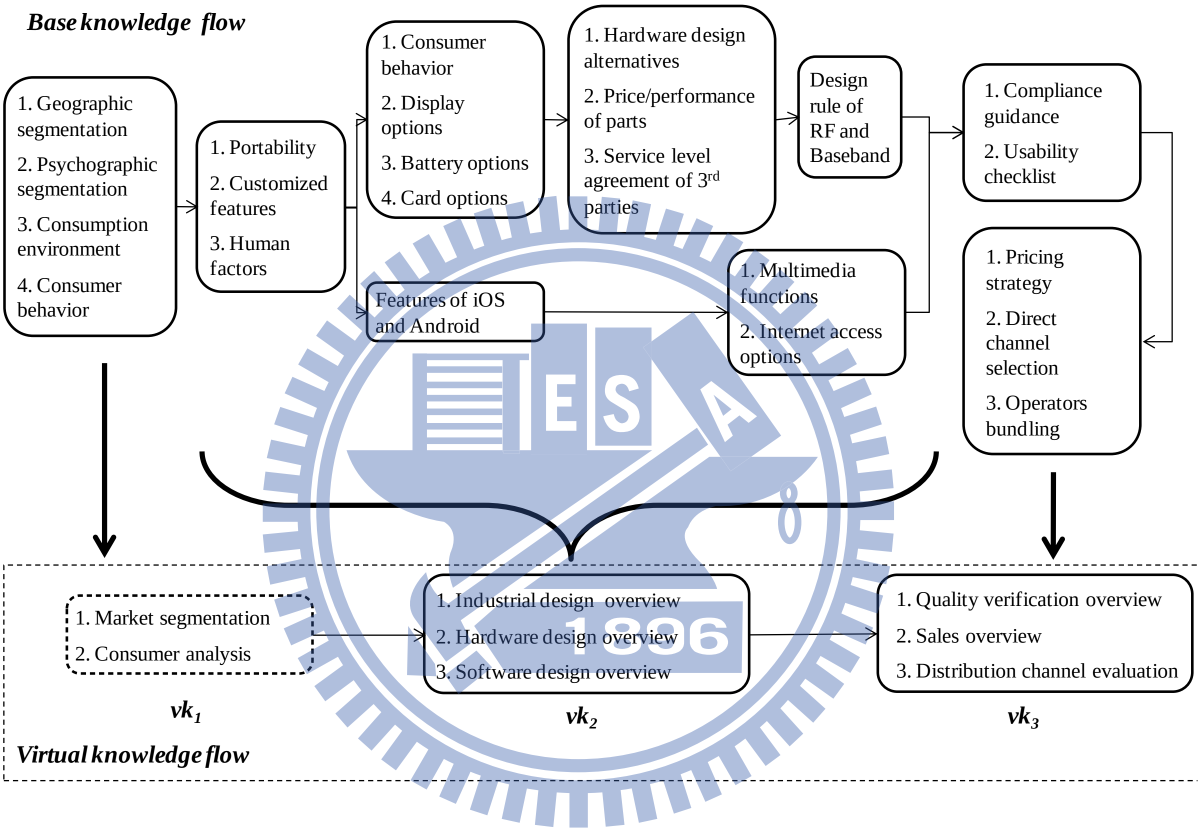


Figure 6. A virtual KF for project managers.

The virtual KF in Figure 6 includes three virtual KNs: vk_1, vk_2 and vk_3, which represent the knowledge-needs of project managers in the mobile phone development process. The virtual KN vk_1 consists of two general knowledge concepts: *market segmentation* and *consumer analysis*, which project managers require to oversee the business analysis task. These two general knowledge concepts are abstracted from four specific knowledge concepts: *geographic segmentation*, *psychographic segmentation*, *consumption environment*, and *consumer behavior*. In node vk_2, which represents product design-related knowledge concepts at a general conceptual level, the three general concepts such as *industrial design overview*, *hardware design overview* and *software design overview* are more helpful to project managers in communicating with product designers than the concepts from which they are abstracted. Finally, the general knowledge concepts in node vk_3 are advantageous to project managers in overseeing verification and commercialization tasks; hence, the virtual KF in Figure 6 appropriately formulates the knowledge-needs of project managers in the development process, and illustrates corresponding knowledge concepts at the proper conceptual levels.

The relationship between base KF and virtual KF can now be described: virtual KFs are the abstracted forms of a base KF. Since virtual KFs are abstractions, different virtual KFs can be generated based on individual participants' knowledge-needs and organization policies. By providing different virtual KFs that hide all or some of the detailed information in a base KF, organizations can be better equipped to enforce policies and fulfill workers' requirements properly. Figure 7 shows an example of mapping a base KF to multiple virtual KFs. While product managers do not need to have detailed knowledge of all the knowledge concepts in the base KF, they must have general marketing knowledge to understand marketing trends and to increase communication effectiveness within a team. To serve product managers' knowledge-needs, knowledge flow designers (KF designers) can abstract marketing-related knowledge nodes and generalize knowledge concepts in those nodes to hide detailed marketing information. A possible virtual KF for product managers is as follows: base KNs k_1 and k_2 are abstracted to virtual KN vk_1, and k_3, k_4, k_5 and k_6 are abstracted to vk_2. In addition, manufacturers have their own virtual KF which contains specific manufacturing knowledge (represented by vk_2), general knowledge of marketing and design (represented by vk_1), as well as general knowledge of sales and post service (represented by vk_3). As this illustrative analysis shows, a base KF can be abstracted to multiple virtual KFs by considering different knowledge-needs and organization policies. In this way, workers can obtain proper virtual KFs that help them acquire the knowledge support they need in collaborative environments.

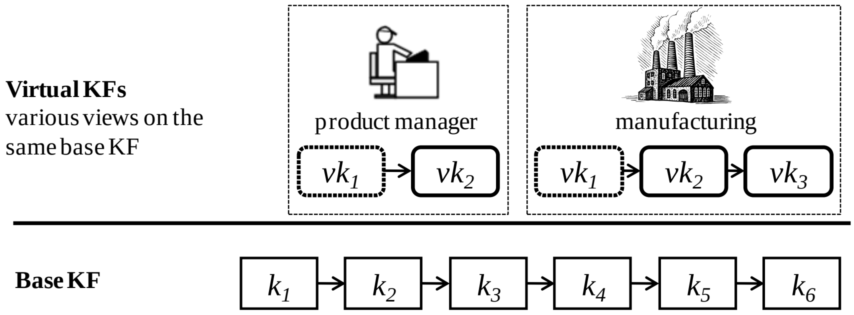


Figure 7. Illustrative examples of base KF and virtual KF.

4.2 The formal framework of the knowledge-flow view (KFV) model

Definition 8 to Definition 13 describes the properties and basic terms that constitute the theoretical framework of the KFV model.

Definition 8: Concealing criteria

A concealing criterion is a 3-tuples <worker *w*, knowledge node *kn*, boolean of abstraction *Y/N* >, which states whether the knowledge concepts of a knowledge node *kn* are too specific or confidential for a worker *w*'s task functions. If the answer is yes, the boolean of abstraction is set to *Y*, and the knowledge concepts of *kn* are abstracted. On the other hand, if the knowledge concepts of *kn* are appropriate for *w*'s task functions and need not be abstracted, the boolean of abstraction is set to *N*.

The concealing criteria are defined by KF designers to comply with company's information security control rules and fulfill team members' need-to-know requirements. KF designers can refer to their experience or utilize experts' knowledge to discover what base KNs should be abstracted if the knowledge concepts in the base KNs are too specific or confidential for workers to perform their task functions properly.

Two scenarios illustrate how KF designers define concealing criteria when deriving virtual KFs from a base KF for a sourcing planner (denoted as *p* for short). The sourcing planner oversees the management of outsourced parts, including: surveying reliable suppliers, evaluating price/performance of parts and negotiating service level agreements with suppliers. The base KF (shown in Figure 3) includes two base KNs, k_2 and k_4, which contain the required knowledge concepts to conduct two tasks: major parts identification task and hardware design task, respectively. In the first scenario, the concealing criterion is made by KF designers, based on information security control rules. In order to sustain competitive advantage, many companies enforce information security policies to protect precious intellectual properties, such as hardware design specifications, from unauthorized access. Only work-related employees can access such specific knowledge or information. The sourcing planner's tasks are not directly related to hardware design tasks, so p is not allowed to access the knowledge concept *design rule of RF and Baseband* which is one type of hardware design specification. Because the knowledge concept *design rule of RF and Baseband* is included in k_4 and p is not allowed to access it, KF designers define a concealing criterion $<p, k_4, Y>$ while deriving virtual KFs for p. The concealing criterion $<p, k_4, Y>$ indicates that k_4's knowledge concept *design rule of RF and Baseband* needs to be abstracted to a general knowledge concept, *IC components*, based on the domain ontology (as shown in Figure 4). The concealing criterion not only protects the specific knowledge concept *design rule of RF and Baseband* from unauthorized access, but also provides general knowledge concept *IC components* for p to effectively communicate with other team members.

In another scenario, the concealing criterion is made in terms of workers' need-to-know requirements. Supposing that the knowledge concepts *consumer behavior*, *display options*, *battery options* and *card options* in k_2 are too specific for p to conduct his/her tasks. Consequently, KF designers define a concealing criterion $<p, k_2, Y>$ to reflect p's knowledge-needs when deriving virtual KFs for p. The concealing criterion $<p, k_2, Y>$ indicates that k_2's knowledge concepts need to be abstracted to the general knowledge concepts *consumer analysis* and *mechanical parts*, respectively, based on the domain ontology (as shown in Figure 4). The two scenarios show that the knowledge for defining concealing criteria is practical and context-dependent, depending on the consideration of security as well as the knowledge-needs of the participants.

Definition 9: Virtual knowledge node

A virtual KN consists of a set of base KNs or previously defined virtual KNs, as well as corresponding knowledge concepts. The knowledge concepts of a virtual KN are abstracted from the knowledge concepts of the corresponding base KNs. A virtual KN *vx* is a 2-tuples <*ANS*, *AKC*>, where *ANS* (Abstracted Knowledge Node Set) is a nonempty set and its members are base KNs or previously defined virtual KNs. *AKC* (Abstracted Knowledge Concept Set) is a nonempty set and its members are knowledge concepts defined in domain ontology.

The knowledge concepts of *vx* are denoted as $AKC(vx) = \{c_1, c_2, c_3, \dots, c_q\}$ where knowledge concept c_i can be identified in domain ontology.

Definition 10: Virtual dependency

Given $BKF=<KNS, DS>$ and two virtual KNs, *vx* and *vy*, a virtual dependency *vdep(vx, vy)* from *vx* to *vy* exists if *dep (x, y)* is in *DS*, where *x* is a member of *vx* and *y* is a member of *vy*. A virtual dependency is used to connect two virtual KNs, *vx* and *vy*.

Definition 11: Virtual knowledge flow

A virtual KF is a 2-tuples, $VKF = <VKNS, VDS>$, where *VKNS* is a nonempty set and its members are virtual KNs, and *VDS* is a nonempty set and its members are virtual dependencies.

Definition 12: Virtual path

Given a virtual knowledge flow $VKF=<VKNS, VDS>$, a virtual path in *VKF*, extending from vk_0 to vk_n, is a sequence of virtual knowledge nodes $vk_0, vk_1, vk_2, \dots, vk_n \in VKNS$, such that $vdep(vk_{i-1}, vk_i) \in VDS$ for $i = 1, 2, \dots, n$. The virtual path from vk_0 to vk_n is denoted as $vk_0 \rightarrow vk_n$.

Definition 13: Virtual ordering relation

Given a virtual knowledge flow $VKF=<VKNS, VDS>$ and two virtual knowledge nodes: *vx* and $vy \in VKNS$, *vx* has a higher order than *vy* if a virtual path $vx \rightarrow vy$ exists. The virtual ordering relation is denoted as $vx > vy$.

Figure 8 illustrates the relationship between the components of the novel model. As the figure shows, a virtual knowledge flow is an abstraction from a base knowledge flow.

The abstraction relationships exist in major components. Virtual knowledge nodes are abstracted from base knowledge nodes; thus, a virtual knowledge node contains generalized knowledge concepts that are abstracted from the knowledge concepts in corresponding base knowledge nodes. Both the abstracted knowledge concepts and the concepts from which they are abstracted can be identified in the domain ontology.

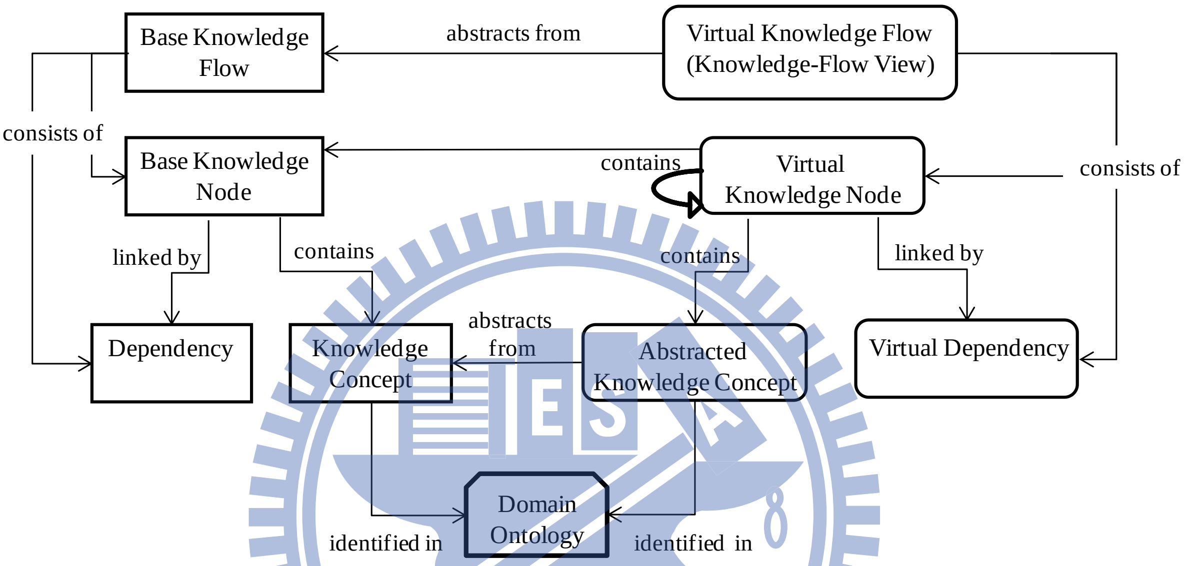


Figure 8. Knowledge-flow view model.

4.3 An order-preserving approach for deriving a knowledge- flow view

Liu et al. [36] presented an order-preserving approach to the generation of virtual processes from a base process in workflow environments. The approach is designed to ensure that the original ordering relation of activities in a base process is preserved in virtual processes. This paper adopts the order-preserving approach for the purpose of generating virtual KFs from a base KF, that retain their knowledge referencing order in the base KF. A legal virtual knowledge node must follow three basic rules to preserve the ordering property in a virtual KF. The basic rules are membership, atomicity and ordering preservation.

Rule 1 (*Membership*): a virtual knowledge node may be abstracted from either base knowledge nodes or previously defined virtual knowledge nodes. The membership among base knowledge nodes and virtual knowledge nodes is transitive. Given three virtual or base knowledge nodes: *x*, *y* and *z*, if *x* is a member of *y* and *y* is a member of *z*, and then *x* is a member of *z*.

Rule 2 (*Atomicity*): a virtual knowledge node is an atomic unit of knowledge access. A virtual knowledge node starts to enable knowledge provision if, and only if, one of its members starts to enable knowledge provision. On the other hand, a virtual knowledge node has stopped its knowledge provision if, and only if, all of its members have stopped their knowledge provision.

Moreover, if an ordering relation (>) between two virtual knowledge nodes exists in a virtual knowledge flow, the implied ordering relation between the respective members of the two virtual knowledge nodes is ">" due to the atomicity rule.

Rule 3 (*Ordering Preservation*): the implied ordering relation between two virtual knowledge nodes' respective members must conform to the ordering relation in the base knowledge flow.

4.4 Procedures for deriving virtual knowledge flows

This section introduces procedures to derive a virtual knowledge flow from a base knowledge flow. Base on concealing criteria and the order-preserving approach, a minimum expanding knowledge node set is formed to determine the member knowledge nodes of a virtual knowledge node. When all virtual knowledge nodes are generated, a procedure of discovering virtual dependencies is applied. Final, a virtual knowledge flow can be generated after the knowledge concepts of virtual knowledge nodes are derived.

4.4.1 Discovering the minimum expanding knowledge node set

KF designers first select some essential knowledge nodes based on team members' knowledge-needs and/or the company's information security control rules to conceal detailed information.

Definition 14: Essential knowledge node

An essential knowledge node is a base knowledge node selected by KF designers for the purpose of generating a virtual knowledge node and generalizing knowledge concepts. To conceal confidential or detailed information, one or more base knowledge nodes in a base knowledge flow should be selected as the essential knowledge node(s).

There are three sets of base knowledge nodes: (a) The Essential Knowledge Node Set (*ENS*) represents the base knowledge nodes selected by KF designers; (b) The Expanding Knowledge Node Set (*ES*) includes the knowledge nodes in *ENS* and the base knowledge nodes which are added due to order-preserving property; and (c) The Neighboring Knowledge Node Set (*NNS*) represents the neighboring (adjacent) base knowledge nodes to the knowledge nodes in *ES*. The knowledge nodes in *NNS* are candidates to be added to the *ES* for preserving the ordering property of a virtual knowledge node. If the implied ordering relation between any knowledge node in *NNS* and any knowledge node in *ES* does not comply with the original ordering relation in the base knowledge flow, the violated knowledge nodes in *NNS* should be incorporated into *ES*. Definition 15 defines a minimum expanding knowledge node set (*MES*) to ensure that only the necessary base knowledge nodes are added, thus preserving the ordering relation while expanding *ES*.

Definition 15: Minimum expanding knowledge node set, *MES*

This set includes both the essential knowledge nodes and the minimum required knowledge nodes which are added to preserve the ordering relation in a virtual knowledge node. The implied ordering relation between any knowledge node in *ES* and any knowledge node not in *ES* must comply with the original ordering relation in the base knowledge flow. Note that an ES_i (a superset of *ENS*) is a *MES* if ES_i satisfies the order-preserving property, and the ES_i does not contain other ES_j (a superset *ENS*) that also satisfies the order-preserving property. The *MES* only contains the essential knowledge nodes and the required knowledge nodes to preserve ordering relations. Based on the *MES*, one can generate virtual knowledge nodes and virtual dependencies, as well as derive knowledge concepts of virtual knowledge nodes.

For a given base knowledge flow, *BKF*=<*KNS*, *DS*>, and an essential knowledge node set, *ENS*, Figure 9 shows the procedure for discovering the minimum expanding knowledge node set *MES*. Initially, the algorithm creates a working set ES_1 of the expanding knowledge node set (*ES*) that initially equals to the essential knowledge node set (*ENS*). According to the ordering preservation rule and *ES* definition, $\forall x \in KNS$, $x \notin ES$, the implied ordering relation between x and all members of *ES* must conform to the ordering relations in the base knowledge flow, *BKF*. *ENS* is the starting point to the discovery of *MES*. A while loop (line 7 to line 10) repeatedly finds any neighboring knowledge node that violates the ordering relation. If a neighboring knowledge node violates the ordering relation conditions (line 9), it is added into *ES*. Finally, $\forall x \in KNS$ and $x \notin ES$; if the implied ordering relations between x and all of the members of *ES* satisfy the ordering preservation, the repeat-until loop stops at this point. The final *ES* is the *MES*, which is the knowledge node set *ANS* of a virtual knowledge node, *vx*, derived from *ENS*.

```
procedure DiscoverMES (INPUT: base knowledge flow $BKF=<KNS, DS>$,
                              essential knowledge node set $ENS$,
                       OUTPUT: virtual knowledge node $vx = <ANS, AKC>$)
begin
    expanding knowledge node set $ES = ENS$
    repeat
        Working Set 1 $ES_1 = ES$
        Neighboring KN Set $NNS = \{x | x, y \in KNS, x \notin ES, y \in ES, dep(x, y) \in DS\}$
        while $NNS$ is not empty do
            Select a knowledge node $x$ from $NNS$ and remove $x$ from $NNS$
            if ($\exists\, y, z \in ES$, such that
                [($x > y$ holds in $KF$) but ($x > z$ does not hold in $KF$)] or
                [($y > x$ holds in $KF$) but ($z > x$ does not hold in $KF$)] then
                Add $x$ into $ES$
            end if
        end while
    until $ES_1 = ES$
    $MES = ES$
    $ANS$ of $vx = MES$
end
```

Figure 9**.** Procedure for discovering the minimum expanding knowledge node set, *MES*.

Furthermore, Figure 10 shows that a base KF includes a loop structure having a single-entry KN (k_1) and a single-exit KN (k_3). Because k_2 and k_4 are identified as essential knowledge nodes and k_2 belongs to the loop structure, KF designers firstly abstract k_2 and other base KNs (k_1, k_3) belonging to the same loop structure to a virtual KN (V_x) as shown in Figure 10(a). Then, KF designers apply the procedure in Figure 9 to discover MES as shown in Figure 10(b) to get final virtual KF as shown in Figure 10(c).

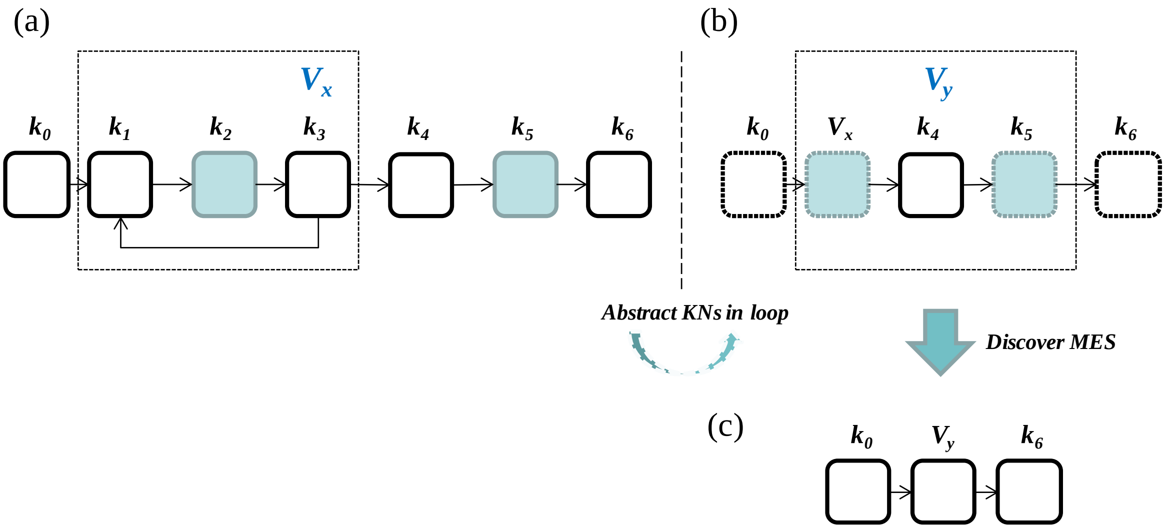


Figure 10. Loop structure in base KF.

4.4.2 Discovering virtual dependencies

All virtual knowledge nodes can be derived from a base knowledge flow *BKF=<KNS, DS>* that form the *VKNS* of a virtual knowledge flow *VKF*, by repeatedly executing the procedure in Figure 9. For any pair of *VKNS*'s members, *vx* and *vy*, the virtual dependency *vdep* (*vx*, *vy*) exists if *dep*(*x*, *y*) exists in *DS*, where *x* is a member of *vx* and *y* is a member of *vy*.

4.4.3 Deriving knowledge concepts of a virtual knowledge node

After a virtual knowledge node, *vx*, has been derived, the knowledge concepts of *vx* should be derived. Figure 11 shows the procedure for deriving the knowledge concepts of a virtual knowledge node. Let *ECS* (Essential Concept Set) denote the set of knowledge concepts of essential knowledge nodes that need to be concealed or hidden. A *minimum generalization policy* is used to generalize (conceal) the concepts in *ECS*. For each concept *c* in *ECS*, the parent concept of *c* in the ontology is selected to form the knowledge concept set (abstracted knowledge concept set, *AKC*) of the virtual knowledge node, *vx*. On the other hand, for some knowledge nodes that are in *MES* but not in *ENS*, the corresponding knowledge concepts do not need to be generalized and are directly included in *AKC* of *vx*. Initially, *AKC* is derived from the generalization of the knowledge concepts in *ECS*. Then, *AKC* incorporates the knowledge concepts of the knowledge nodes in *MES* but not in *ENS*. If some knowledge concepts in *AKC* are members of *ECS* or the descendant concepts of *ECS*’s members, they are removed from *AKC* due to the concept concealing requirement. The final step is to remove the implied (redundant) concepts from *AKC*; hence, the knowledge concepts of *vx* can be obtained.

If KF designers want to add or delete knowledge concepts in a virtual knowledge flow, it is appropriate to do these operations in the corresponding base knowledge flow and then re-generate a new virtual knowledge flow to replace the old one. A virtual knowledge flow is derived from a base knowledge flow. Conceptually, it is difficult to map back any changes in a virtual KF to the corresponding base KF. Thus, these operations should be made in the base KF. A similar concept exists in database management systems, where administrators modify the definition of a database view by adding or deleting fields in the underlying base tables.

```
procedure DeriveKnowledgeConcepts
INPUT: minimum expanding knowledge node set $MES$,
       essential knowledge node set $ENS$
OUTPUT: virtual knowledge node $vx = <ANS, AKC>$
begin
    Essential Concept Set $ECS$ = empty set
    $AKC(vx)$ = empty set
    for each knowledge node $x$ in $ENS$ do $ECS = ECS \cup KC(x)$
    for each knowledge concept $c$ in $ECS$
        do $AKC(vx) = AKC(vx) \cup \{Generalization(c)\}$
    for each knowledge node $y$ in $MES$ but not in $ENS$ do $AKC(vx) = AKC(vx) \cup KC(y)$
    for each knowledge concept $c$ in $AKC(vx)$ do
        if $c$ is in $ECS$ or $c$ is a descendant concept of $ECS$'s member then
            remove $c$ from $AKC(vx)$
        for each knowledge concept $c$ in $AKC(vx)$ do
            if $c$ is an implied concept under $AKC(vx)$ then remove $c$ from $AKC(vx)$
        return AKC(vx)
end
```

Figure 11. Procedure for deriving knowledge concepts of a virtual knowledge node.

Definition 16: Implied concept

A concept c is implied under a concept set C if c can be inferred by other concepts in C. Based on a domain ontology, the concept c is mapped to an ontology node k that has n child ontology nodes k_i (i=1 … n). The concept c is an implied concept if each k_i's corresponding concept is either in concept set C or can be implied by other concepts in concept set C.

4.5 Case illustration and analysis

This section uses a base knowledge flow of a mobile phone company, named Smart-Tech Company, to illustrate the application of the KFV model and conduct preliminary analysis. The base KF represents the knowledge-needs that a project team requires when conducting a mobile phone development process in the company. According to the process, KF designers consult domain experts and team participants to acquire important knowledge-needs and identify corresponding knowledge concepts for the purpose of representing knowledge-needs in a base KF.

In this company, the mobile phone development team requires participants from various departments. Those team members have different task functions: (1) the project manager controls and coordinates the project, (2) the marketing analyzer conducts the business analysis, (3) the designer is responsible for product design, (4) the salesperson focuses on product commercialization, (5) the inspector carries out the quality assurance tasks, and (6) the sourcing planner oversees the management of outsourced parts. Based on different knowledge-needs, KF designers can design virtual KF for individual participants.

The following discussion pertains to the sourcing planners at this company whose task function is parts outsourcing. First, KF designers make the concealing criteria for the sourcing planners, as required by the information security policy of the company and in consideration of the information granularity suggested by domain experts. Hence, the *Essential Knowledge Nodes* are identified based on the concealing criteria and all knowledge concepts in the *Essential Knowledge Nodes* should be included in an *Essential Concept Set*. Then, a virtual knowledge node is obtained by the order-preserving approach to ensure that the ordering in the base knowledge flow is retained. Finally, the KF designers abstract the knowledge concepts in the *Essential Concept Set* using the domain ontology and the minimum generalization policy.

The knowledge flow in Figure 12 includes nine knowledge nodes, k_0 to k_8, where each knowledge node contains multiple knowledge concepts. The knowledge concepts of k_2 are too specific for the sourcing planners, so the KF designers make a concealing criterion <*sourcing planner*, k_2, *Y*> to meet their knowledge-needs. Another concealing criterion <*sourcing planner*, k_4, *Y*> is also made because the knowledge concepts of k_4 are confidential for the sourcing planners. Following the two concealing criteria, the KF designers select two knowledge nodes, k_2 and k_4, as the essential knowledge nodes for the sourcing planners.

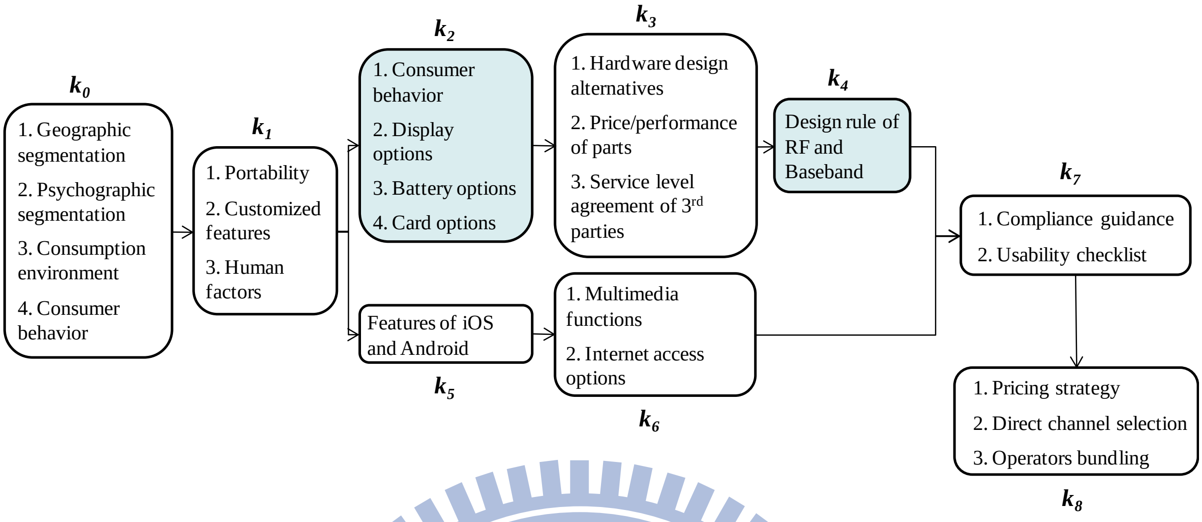


Figure 12. Base KF for sourcing planners (nodes k_2 and k_4 are essential KNs).

The KF designers apply the procedure in Figure 9 to obtain a virtual knowledge node. Initially, the neighboring knowledge node set $NNS = \{k_1, k_3, k_7\}$ and ENS (essential knowledge node set) = ES (expanding knowledge node set) = $\{k_2, k_4\}$. Knowledge node k_3 is added into ES since the ordering of k_2 is higher than the ordering of k_3 (i.e. $k_2 > k_3$), but the ordering of k_4 is not higher than the ordering of k_3 (i.e. $k_4 \not> k_3$). Knowledge node k_7 is not added into ES since $k_2 > k_7$ and $k_4 > k_7$. Knowledge node k_1 is not added into ES since $k_1 > k_2$ and $k_1 > k_4$. Therefore, ES is changed to $\{k_2, k_3, k_4\}$. In the second execution, $NNS = \{k_1, k_7\}$ and $ES = \{k_2, k_3, k_4\}$. Knowledge node k_1 and knowledge node k_7 are not added into ES because the implied ordering relations between each member in NNS and ES satisfy the ordering preservation rule. Therefore, the execution stops. The minimum expanding knowledge node set MES includes knowledge nodes $\{k_2, k_3, k_4\}$, and a virtual knowledge node vk_1 is derived as shown in Figure 13.

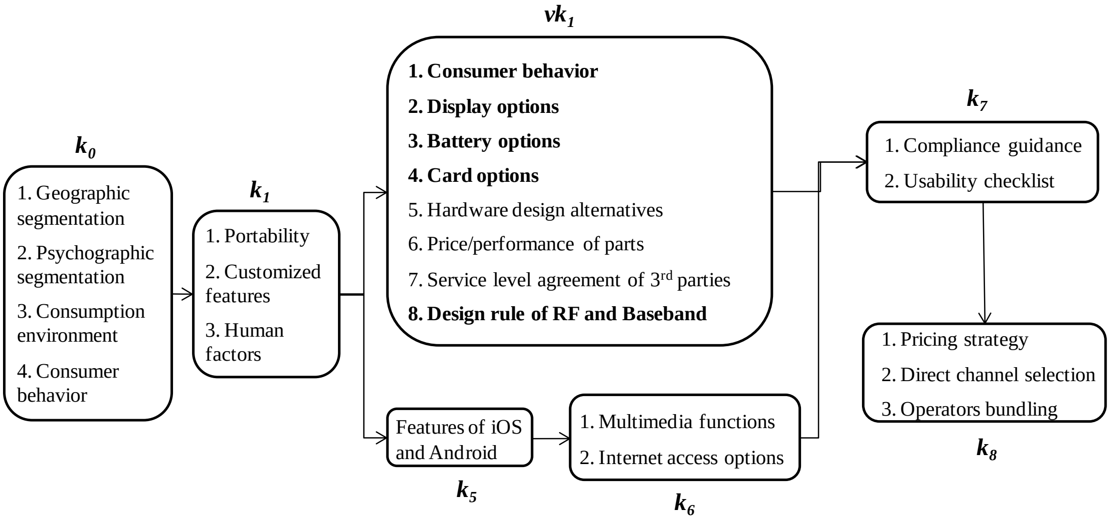


Figure 13. A virtual KN, vk_1, obtained after applying order-preserving approach.

After discovering the minimum expanding knowledge node set *MES*, the KF designer uses the procedure in Figure 11 and the ontology in Figure 14 to derive the knowledge concepts of vk_1 based on the minimum generalization policy.

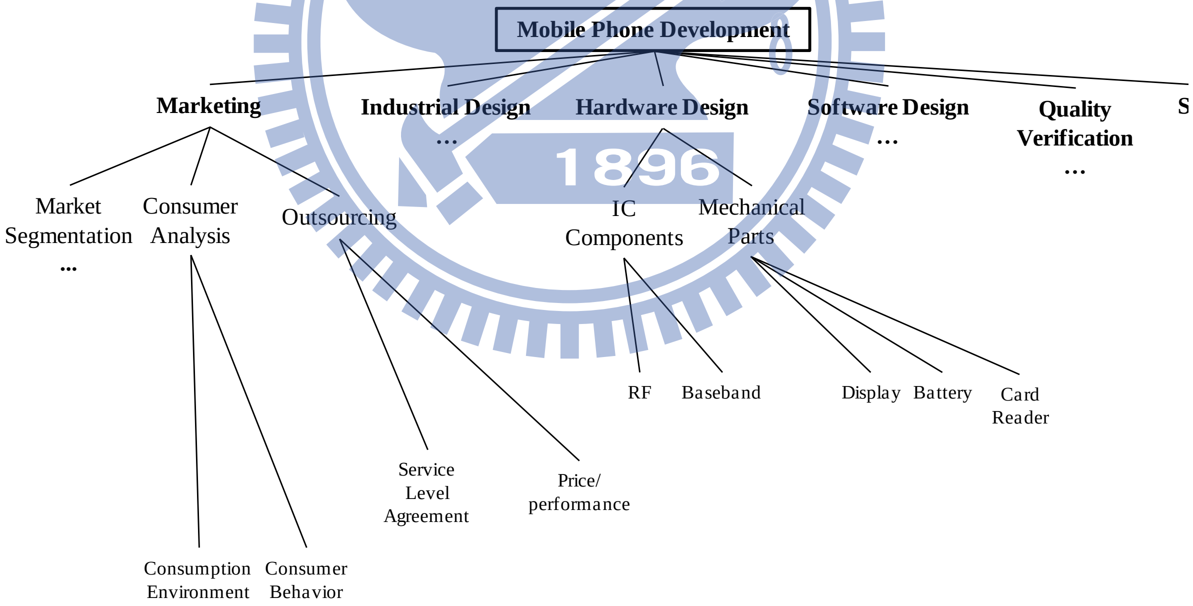


Figure 14. A partial domain ontology of mobile phone development.

Initially, the essential concept set *ECS* equals to {*consumer behavior*, *display options*, *battery options*, *card options*, *design rule of RF and Baseband*}. After performing the *Generalization* function on *ECS*, the KF designer obtains the abstracted knowledge concept set, *AKC* = {*consumer analysis*, *mechanical parts*, *IC components*}. Then, the knowledge concepts *hardware design alternatives*, *price/performance of parts* and *service level agreement of 3rd parties* are added in *AKC*, since knowledge node k_3 is in *MES* but not in *ENS*. Hence, *AKC* = {*consumer analysis*, *mechanical parts*, *IC components*, *hardware design alternatives*, *price/performance of parts*, *service level agreement of 3rd parties*}. The knowledge concepts of vk_1 can thus be obtained, as shown in Figure 15.

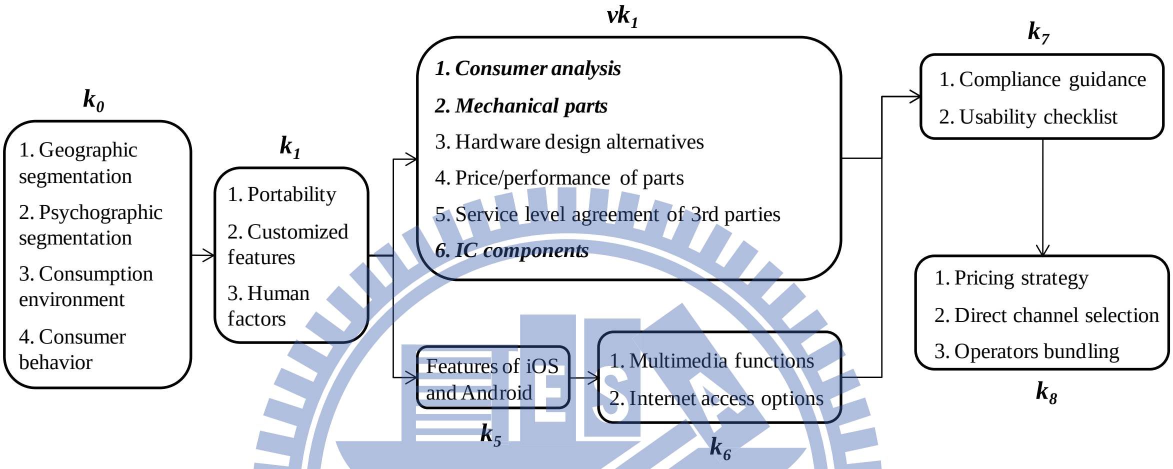


Figure 15. Knowledge concepts of vk_1 after applying the minimum generalization policy.

Figure 15 show that vk_1 has a redundant knowledge concept, *hardware design alternatives*, that can be inferred by *mechanical parts* and *IC components*. Figure 16 shows the result after removing the concept *hardware design alternatives*. The final knowledge concepts of vk_1 are: *consumer analysis*, *mechanical parts*, *IC components*, *price/performance of parts* and *service level agreement of 3rd parties*.

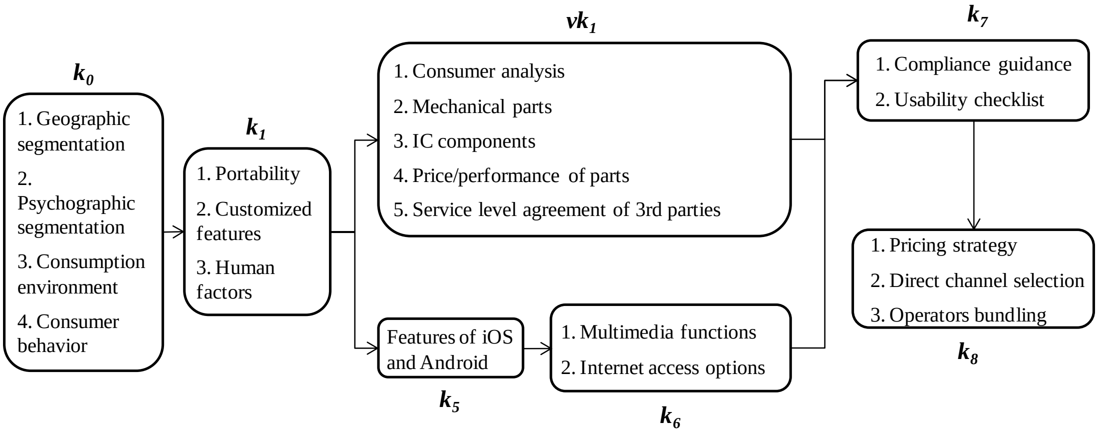


Figure 16. Knowledge concepts of vk_1 after removing implied knowledge concept.

The example given demonstrates the virtual knowledge flow from the sourcing planners' perspective. If the sourcing planners are not satisfied with the view that has been generated, KF designers can repeat the same steps to abstract a new virtual knowledge flow by identifying other knowledge nodes as essential knowledge nodes, and then generating new MES to form another virtual knowledge node. Similarly, it is possible to create other virtual knowledge flows from other members' perspectives. Hence, the proposed KFV model can enhance conventional knowledge flow models by supporting different team members with various knowledge-needs. Finally, every team member can obtain a proper virtual knowledge flow to support his/her knowledge-needs in the collaborative knowledge support platform.

To test the practical implications of this study, a preliminary analysis was conducted. Several professionals were invited to examine the case and related concepts to investigate whether the theoretical model could benefit them. Overall, there was general agreement regarding the feasibility of the KFV model and its practical value. They thought that the KFV model could enhance typical knowledge flows to serve all team participants with their various knowledge-needs. For example, an interviewee mentioned that he would be able to communicate with a hardware designer more efficiently because the ontology of the theoretical model provided a common understanding of the general knowledge of hardware design. By referring to their different knowledge-flow views, both participants would be able to better understand their different knowledge-needs. Such understanding has the potential to improve the quality of communication and increase the efficiency of knowledge sharing.

The results of the preliminary analysis are summarized as follows: (1) the visualized knowledge flows and knowledge-flow views help team members to easily formulate their knowledge-needs and quickly obtain consensus under common domain ontology. Thus, the quality of their communication and decision making is improved; (2) knowledge-flow designers can produce concealing criteria to protect confidential knowledge from unauthorized access and solve the information overload problem by abstracting detailed knowledge; and (3) in organizations, knowledge-flow views extend the efficiency of knowledge flows and improve the effectiveness of knowledge sharing and knowledge support.

4.6 Discussion

This chapter investigates the shortage of knowledge support in collaborative teams. The workers in a team usually have different knowledge-needs according to their task functions. For example, the mobile phone development process involves six task functions. The workers in these task functions need to access different knowledge concepts at different conceptual levels to conduct their work and communicate with each other. However, conventional knowledge flow models do not provide different views of a knowledge flow that are required to address individual needs. The proposed KFV model meets this and related challenges. According to the proposed model, KF designers select some base knowledge nodes from a base knowledge flow to generate virtual knowledge nodes that conceal confidential or detailed information. Through an order-preserving approach and a knowledge concept generalization mechanism, a virtual knowledge flow is generated. The proposed innovative model allows various virtual knowledge flows to be generated that meet the individual knowledge-needs of different workers. These virtual knowledge flows not only comply with organizational information security policy but also reflect the granularity of knowledge-needs. Thus, this study advances the conceptual applicability of knowledge flow research to cooperative knowledge support environments.

Practical implications can be derived from the KFV model, including knowledge support facilitation in cooperative teams and team productivity and communication quality improvement. Moreover, the KFV model can be applied to any knowledge-based organization where business processes are conducted by cooperative teams in a dynamic working environment.

Chapter 5 Role-based knowledge-flow view model

According to the knowledge-flow view (KFV) model described in Chapter 4, knowledge-flow designers (KF designers) identify essential knowledge nodes based on concealing criteria and abstract base knowledge nodes (base KN) to virtual knowledge node (virtual KN) to build virtual knowledge flows (virtual KF) for the purpose of facilitating cooperative knowledge support. Except considering concealing criteria, tasks are often assigned to dedicated roles based on the characteristics of tasks to ensure quality and security. Hence, workers have different knowledge-needs in terms of roles. This leads to develop a role-based knowledge-flow view (r-KFV) model to illustrate role-based knowledge-needs properly in teamwork environments. This chapter extends the KFV model to the r-KFV model by introducing a role-based framework and role-based knowledge flow abstraction methods.

The r-KFV model analyzes the levels of knowledge required by workers based on their roles, and develops role-based knowledge flow abstraction methods that generate virtual knowledge nodes to provide the appropriate level of knowledge for each role. The purpose of the r-KFV model is to derive role-based virtual knowledge flows (VKFs) from a base knowledge flow (BKF). In the model, virtual knowledge nodes (virtual KNs) are generated from base knowledge nodes (base KNs) based on the relevance degrees between roles and base KNs. In addition, a concept abstraction method is developed to abstract knowledge concepts of base KNs for a virtual KN.

5.1 Concepts of role-based virtual knowledge flows

This section introduces a role-based framework in the r-KFV model and presents the key concepts of role-based virtual knowledge flows.

A role-based virtual knowledge flow comprises a set of virtual KNs that are aggregated from base KNs according to their relevance to a role. Some base KNs may be more relevant to a role than others. In other words, a role may refer to some knowledge nodes frequently but refer to others rarely due to its responsibilities and authorities. The relevance degrees of a role to base knowledge nodes are distinct and indicate how important the base knowledge nodes to the role are. A virtual KN of a role-based virtual KF denotes a meaningful knowledge unit of interest to the role; thus, it should be relevant to the role. The process of identifying a role-based virtual KN involves aggregating base KNs based on their relevance to the role.

Once the relevance of the aggregated knowledge nodes to the role reaches a certain threshold, a virtual knowledge node is identified for the role. The relevance of a knowledge node to a particular role can be specified by KF designers or derived from the relevance degrees between the role and the operations associated with the knowledge nodes. Based on the relevance degrees of all base knowledge nodes, procedures can be clearly defined to generate appropriate role-based virtual KFs for different organizational roles.

After a virtual KN has been identified, KF designers can derive its knowledge concepts. The objective is to obtain abstractions of the knowledge concepts in the virtual KN's member knowledge nodes that do not conform to the knowledge required by the role. The role-based knowledge requirements are defined through domain ontology, which is a hierarchy comprised of knowledge concepts. The lower levels of the hierarchy contain specific knowledge, while the upper levels contain knowledge that is more general. The various roles in a team have different knowledge requirements. Participants need specific knowledge about their own roles and tasks, but needs less specific knowledge about other roles' tasks. For example, Workers with a researching role design and develop products, so they must have specific technical skills and knowledge. Worker with a marketing role, who launch and promote products, may not have specific technical knowledge about the products, but they must have general knowledge about the technical aspects. The granularity of knowledge such as general or specific can be represented by knowledge concept levels in domain ontology. Consequently, building virtual knowledge flows from a roles perspective and considering the granularity of knowledge are necessary to illustrate knowledge-needs properly in teamwork environments.

Definition 17: Concept level, *CL*

A knowledge concept is mapped to the corresponding concept level in domain ontology. The concept level is defined by letting the root of the domain ontology be level one. If a knowledge concept is at level l, then its child concepts are at level $l + 1$. The concept levels indicate the levels of knowledge concepts and represent their granularity; that is, the knowledge concepts with larger concept levels (i.e., in the lower levels of domain ontology) are more specific than those with smaller concept levels (i.e., in the upper levels of domain ontology). A knowledge concept's generality (or specificity) is determined by its concept level.

5.2 A role-based framework

The crucial part of deriving role-based virtual KFs is to find the relevance degree between each knowledge node and each role in a given base KF. The relevance degrees can be specified by KF designers. But, it's time consuming and complex for them to manually determine the degrees for each role. Hence, a role-based framework is utilized to calculate the relevance degrees by a derivation process. Based on the derived relevance degrees, a proposed approach can then generate appropriate virtual KFs for different organizational roles. The derivation process of generating a role-based virtual KF incorporates the concept of operations, which is essential in organizational environments. In this section, the concepts of role-operation relevance profiles and operation knowledge requirements are proposed in the role-based framework.

The domain ontology defined in Definition 1 can be further divided into multiple knowledge categories. Workers can express their knowledge requirements in terms of knowledge categories in the role-based framework. Figure 17 shows the domain ontology of mobile phone development comprising of multiple knowledge categories. The *marketing* knowledge category and *hardware* knowledge category are highlight in blocks. The corresponding concept levels of knowledge concepts are also identified in the figure. For example, the concept levels of *marketing* and *consumer analysis* are 2 and 3 respectively.

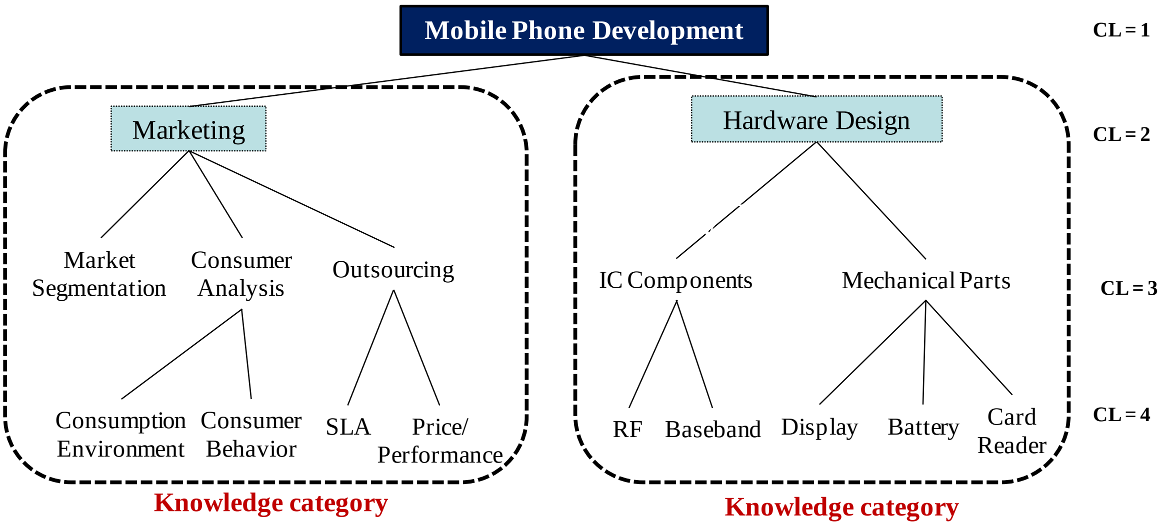


Figure 17. Partial domain ontology with knowledge categories and concept levels.

Each base KN is associated with a set of operations that participating roles may perform. The relevance degree is used to determine how relevant an operation is to a role. The relevance degree is used as a quantified value to abstract base KNs into virtual KNs.

Definition 18: Base knowledge node profile

A base KN is associated with different operations. The base knowledge node profile is a set of 2-tuple <base knowledge node *kn*, operation *op*>, which defines operation *op* is associated with base knowledge node *kn*.

Definition 19: Role-operation relevance profile

The profile, which records the relevance degree between a role and an operation, is a set of 3-tuple <role *r*, operation *op*, operation relevance degree *ordeg* >. Each 3-tuple records that the operation relevance degree between role *r* and operation *op* is *ordeg*. Roles have different degrees of relevance to their authorized operations. The more the relevance between a role and an operation, the higher *ordeg* will be. The *ordeg* is limited to the range [0, 1].

Definition 20: Operation required knowledge concept profile

An operation required knowledge concept profile comprises a set of 3-tuple <operation *op*, knowledge category *ca*, knowledge concepts *kcs*>, which represents the required knowledge concepts *kcs* in knowledge category *ca* for performing operation *op*.

The domain ontology is divided into knowledge categories, and an operation may be related to more than one knowledge category. A knowledge concept set indicates which knowledge concepts in a given knowledge category are required to perform an operation.

Definition 21: Role-operation knowledge requirement degree profile

For a given operation, different roles may require different degrees of knowledge in different knowledge categories. A role-operation knowledge requirement degree profile is a set of 4-tuple <role *r*, operation *op*, knowledge category *ca*, knowledge requirement degree *krdeg*>. Each tuple indicates that the degree of knowledge required by role *r* with respect to operation *op* in knowledge category *ca* is *krdeg*. The knowledge requirement degree is used to define how specific or general the required knowledge in a certain knowledge category should be for role *r* while executing operation *op*.

Given a sourcing department manager role *r* and two operations related to the part sourcing task in Figure 2: evaluating display suppliers (op_1) and surveying battery suppliers (op_2), Figure 18 shows a role-based framework applied in the context. The role-based framework describes the relationships among role, operation and knowledge node which are the basic concepts for generating a role-based virtual KF. Overall speaking, two phases are required to generate virtual KFs. The two phases are mutually independent. Phase I generates virtual KNs and Phase II derives knowledge concepts for these virtual KNs. In Figure 18, the role-operation relevance profile, base knowledge node profile and threshold *TH* are used to generate virtual KNs in Phase I; the role-operation knowledge requirement degree profile and operation required knowledge concept profile are utilized to abstract knowledge concepts for virtual KNs in Phase II.

In this example, the relevance degrees *ordeg* of *r* to operation op_1 and op_2 are 0.2 and 0.05 respectively (shown in role-operation relevance profile)**;** op_1 and op_2 is associated with kn_1 (shown in base knowledge node profile); the knowledge concepts required in the *marketing* category to perform op_1 are {*c131*, *c132*} and in the *hardware* category is {*c621*} (shown in operation required knowledge concept profile); the knowledge requirement degree *krdeg* of role *r* to perform op_1 is 0.8 for the *marketing* category and 1.0 for the *hardware* category (shown in role-operation knowledge requirement degree profile). The above information is used to generate a role-based virtual KF as well as derive the corresponding knowledge concepts for the virtual KNs in the virtual KF.

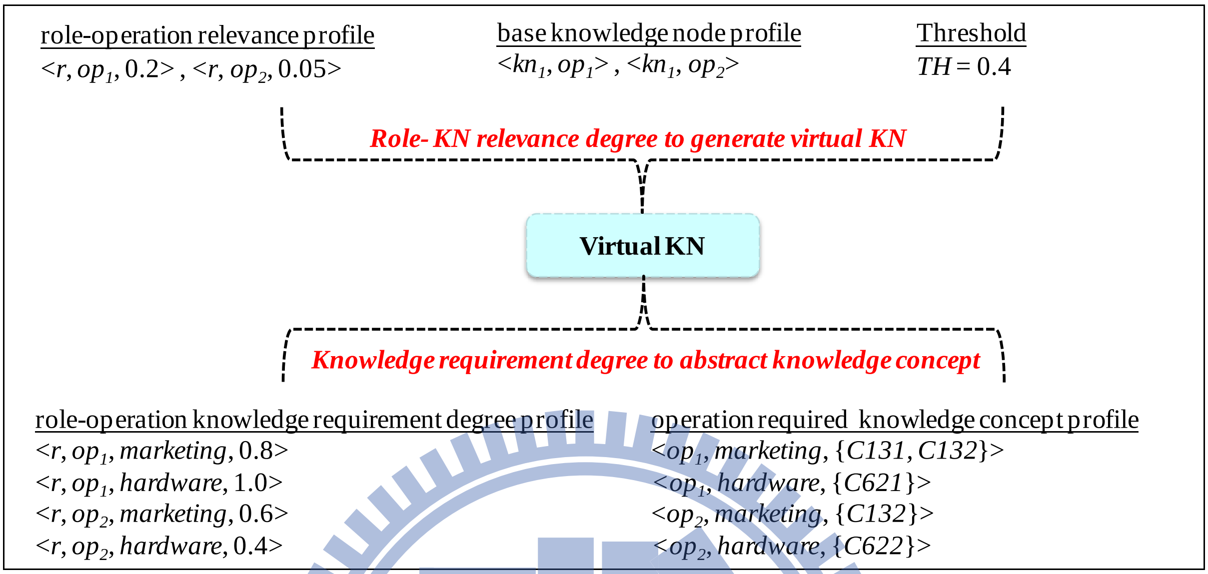


Figure 18. A role-based framework with examples.

5.2.1 Construction of role-operation relevance profile

Initially, role-operation relevance profiles are constructed by analyzing a role's operating logs. Let T denote the number of times that role r performs all of the assigned operations in a period, and let N denote the number of times that role r performs an operation op in the same period. The default operation relevance degree of r with respect to op is N/T. Take Figure 18 as an example, role r performs op_1 four times, op_2 once, and other operations 15 times; then T equals *20*. The operation relevance degree *ordeg* of r is 0.2 for op_1 and 0.05 for op_2 as shown in Figure 18. Intuitively, a higher *ordeg* indicates greater relevance between a role and an operation. Moreover, the higher the cost associated with an operation performed by a role, the higher the relevance degree of the operation to the role will be. Let Q denote the total cost of the operations assigned to role r, and let C denote the cost of a specific operation op. The *ordeg* of role r when performing operation op is C/Q. Based on activity-based costing (ABC) models, the cost can be measured in terms of the time and resources expended by roles when they perform assigned operations. In summary, different statistics can be extracted from the historical log data. Some decision-making methods can be employed to derive the *ordeg* by combining the statistics.

It is noteworthy, while implementing a role-based KFV system based on the r-KFV model, that the proposed methods and parameters for constructing the role-operation relevance profiles need to be fine-tuned in terms of the culture of organizations, the accommodation of peripheral systems and the context of operations. Workers' future operation logs can also be used to adjust *ordeg* to satisfy their real time knowledge-needs. These adjustments are essential to obtain appropriate virtual knowledge flows.

5.2.2 Evaluation of role-knowledge node relevance

The relevance degree between a given role and a base knowledge node can be derived from the role-operation relevance profiles and base knowledge node profiles. For example, role-operation relevance profile {< r, op_1, 0.2 >, < r, op_2, 0.05 > …} and a knowledge node profile {< kn_1, op_1>, < kn_1, op_2>, …}, the relevance degree between role r and base knowledge node kn_1 is max (0.2, 0.05, …) = 0.2. That is, the relevance degree between role r and base knowledge node kn_1 is the maximum of the *ordeg* of multiple operations that r is authorized to refer to base knowledge node kn_1.

5.3 Procedures for deriving role-based virtual knowledge flows

This section presents the approach for discovering role-based virtual KFs suitable for participating roles through applying the role-based framework. The approach involves three steps: (a) identifying role-based virtual KNs by aggregating base KNs based on their relevance to a role, (b) building virtual dependencies to connect the identified virtual KNs, and (c) deriving knowledge concepts of the identified virtual KNs based on the operation required knowledge concepts and role's operation knowledge requirement degrees.

5.3.1 Identifying role-based virtual knowledge nodes

The identification procedure in Figure 19 starts from taking the highest ordering base KNs from a base KF as a seed node. Beginning with the seed, the procedure repeatedly aggregate the adjacent base KNs of the seed according to the descending order of their role-knowledge node relevance degrees. Until the total relevance degree of the aggregated base KNs approximates a granular threshold, a virtual KN is identified. Other base KNs which are not aggregated forms a residual knowledge node set (*RKNS*). The base KNs in *RKNS* mean that they are not the member knowledge nodes of any virtual KNs yet. The procedure takes the highest ordering base KN from *RKNS* as another seed node and repeats another aggregation run. The loop continues until no more base KNs in *RKNS*. Hence, all role-based virtual KNs have been generated. Then, the virtual dependencies among the identified virtual KNs can be set as described in Section 4.4.2.

Moreover, during the aggregating loop, the order-preserving property should be checked. The order-preserving approach discussed in Section 4.1.1 is applied to ensure that a role-based virtual KF maintains the knowledge referencing order in it as the order in the corresponding base KF. The detail algorithm of identifying role-based virtual KNs is described in a previous work [35].

```
procedure Identify Virtual KN (input: base KF, role $r$, threshold $TH$, output: virtual KF)
begin
  put waiting-for-aggregated base KNs to RKNS (residual KN set)
  repeat
    select the highest order base KN $x$ from RKNS
    /* generate a role-based virtual KN which consists of $x$ */
    add a neighbor of $x$ with the largest relevance degree to form temp virtual KN
    while (temp virtual KN statisfy
          Order-preserving
          $F_{TRD}$ ($r$, temp virtual KN) <= $TH$
          No overlap base KN)
    do (greedily include additional neighboring KN with the largest relevance degree
       and apply order-preserving approach to add relevant KNs to form temp virtual
       KN)
    generate a role-based virtual KN according to prior temp virtual KN
    put the generated virtual KN in virtual KF
  until no more base KNs in RKNS
```

```
    set virtual dependencies for all identified virtual KNs
end
```

Figure 19. Procedure for identifying role-based virtual KNs.

Definition 22: Total relevance degree, F_{TRD}

Let a function f_{KRD} (role r, base knowledge node kn) return the relevance degree of kn to role r; and let a function F_{TRD} (role r, *base* knowledge node set V) return the total relevance degree of a virtual knowledge node *vkn* with knowledge node set V. $F_{TRD}(r, V) = \Sigma f_{KRD}(r, kn_i)$ for all $kn_i \in V$. The total relevance degree estimates the closeness between a role and a set of base knowledge nodes.

5.3.2 Deriving Knowledge Concepts of Virtual Knowledge Nodes

Deriving knowledge concepts for a virtual KN, *vkn*, is to abstract all knowledge concepts of its member knowledge nodes to proper concept levels based on role *r*'s knowledge requirement degrees. An illustration example is shown in Figure 20.

Supposedly, the *vkn* consists of two base KNs, k_1 and k_2, as shown in Figure 20(a). The knowledge concepts of k_1 and k_2 should be abstracted. The knowledge concept abstraction process involves following steps and utilizes profiles and settings in the role-based framework:

(1) Identify the set of operations performed by r and associated with *vkn*. The set OP_r^{vkn} can be obtained from combining knowledge node profiles of *vkn*'s member KNs. According to Figure 20(b), OP_r^{vkn} is $\{op_x, op_y\}$ after combing knowledge node profiles $\{<k_1, op_x>, <k_2, op_y>\}$.

(2) Discover the set of required knowledge concepts to conduct operations $op \in OP_r^{vkn}$. The set KCS_{op}^{ca} comprises a set of 3-tuple <operation *op*, knowledge category *ca*, knowledge concepts *KCS* >. It illustrates that the knowledge concepts *KCS* in *ca* are required to perform *op*. KCS_{op}^{ca} can be obtained from operation required knowledge concept profile.

Based on the operation required knowledge concept profile shown in Figure 20(d),

KCS_{op}^{ca} can be formed as { <op_x, ca_1, {C_{111}, C_{112}}>, <op_y, ca_1, {C_{112}}>, <op_x, ca_2, {C_{221}}>, <op_y, ca_2, {C_{222}}>}.

(3) Let CRD_r^{vkn} be a set of 3-tuple <concept *c*, category *ca*, knowledge requirement degree *krdeg* >, which indicates the knowledge requirement degree *krdeg* of knowledge concept *c* for role *r* with respect to knowledge category *ca*. CRD_r^{vkn} is derived from KCS_{op}^{ca} and role-operation knowledge requirement degree profile.

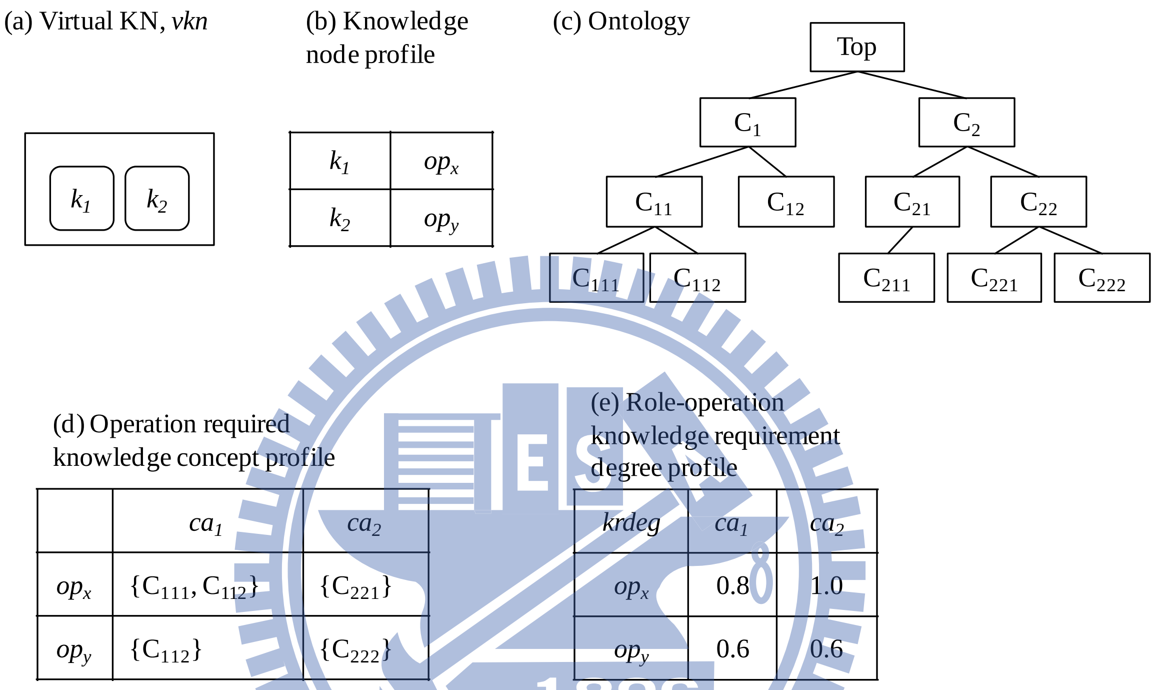
Based on Figure 20(e), the role-operation knowledge requirement degree $krdeg_{r,op}^{ca}$ is { <op_x, ca_1, 0.8>, <op_y, ca_1, 0.6>, <op_x, ca_2, 1.0>, <op_y, ca_2, 0.6> }, that is, $krdeg_{r,op_x}^{ca_1}$ =0.8; $krdeg_{r,op_y}^{ca_1}$ =0.6; $krdeg_{r,op_x}^{ca_2}$ =1.0 and $krdeg_{r,op_y}^{ca_2}$ = 0.6. Combining KCS_{op}^{ca} and $krdeg_{r,op}^{ca}$, the knowledge requirement degree $krdeg_{r,c}^{ca}$ for each knowledge concept can be obtained: $krdeg_{r,c_{111}}^{ca_1} = krdeg_{r,op_x}^{ca_1} = 0.8$; $krdeg_{r,c_{112}}^{ca_1}$ = maximum ($krdeg_{r,op_x}^{ca_1}$, $krdeg_{r,op_y}^{ca_1}$) = maximum (0.8, 0.6) = 0.8; $krdeg_{r,c_{221}}^{ca_2} = krdeg_{r,op_x}^{ca_2} = 1.0$; $krdeg_{r,c_{222}}^{ca_2} = krdeg_{r,op_y}^{ca_2} = 0.6$. Noteworthy, the knowledge concept c_{112} is required by two operations, op_x and op_y. Thus, the knowledge requirement degree of c_{112} ($krdeg_{r,c_{112}}^{ca_1}$) is the maximum of $krdeg_{r,op_x}^{ca_1}$ and $krdeg_{r,op_y}^{ca_1}$.

(4) Adjust (generalizing) the knowledge concepts to the appropriate concept level according to the role's knowledge requirement degree $krdeg_{r,c}^{ca}$. The knowledge requirement degrees are used to abstract the required knowledge concepts into appropriate concept levels in order to satisfy a role's operation knowledge requirement. The adjust step follows the hierarchy structure of domain ontology to generalize knowledge concepts to their parent knowledge concept until the concept levels of the parent knowledge concepts meets roles' requirement $krdeg_{r,c}^{ca}$.

A function *GenACL* is defined to map the knowledge requirement degree to the appropriate adjusted concept level. *GenACL* may be defined to map the value of degree 0.8 to one concept level of abstraction, and the value 0.6 to two levels of abstraction, and so on. For example, the concept level *cl* of c_{111} is 4 based on the ontology shown in Figure 20(c). The knowledge requirement degree of c_{111} ($krdeg_{r,c_{111}}^{ca_1}$) is 0.8. Thus, *GenACL* (*cl*, *ca*, $krdeg_{r,c}^{ca}$) = *GenACL* (*4*, ca_1, *0.8*) = 3. It means that c_{111} should be abstracted to its parent concept (c_{11}) at level 3 in ca_1 by applying function *GenConcept*.

(5) Follow Definition 16 (Implied Concept) to remove the implied (redundant) concepts.

Iteratively applying the five steps to all virtual KNs in the deriving virtual knowledge flow, the virtual knowledge flow should be proper for the role *r* by fulfilling his/her knowledge requirements.



k_1	op_x
k_2	op_y

	ca_1	ca_2
op_x	$\{C_{111}, C_{112}\}$	$\{C_{221}\}$
op_y	$\{C_{112}\}$	$\{C_{222}\}$

krdeg	ca_1	ca_2
op_x	0.8	1.0
op_y	0.6	0.6

Figure 20. Illustration example of deriving knowledge concepts for a virtual KN.

The detail algorithm of deriving knowledge concepts of role-based virtual KNs is described in a previous work [35].

5.4 Designing a role-based KFV system

This section analyzes the proposed r-KFV model and concepts to conduct the basic system design of a role-based KFV system. The basic system design constructs an overall architecture by identifying important functional modules and decomposing them into layers. An activity diagram models the procedural flow of actions from the perspectives of KF designers, roles and experts. Moreover, Protégé 4.0 software is used to build an ontology prototype to represent knowledge concepts and their hierarchical relationships in the mobile phone development domain. The system architecture, activity diagram and ontology prototype are the fundamental elements for putting the theoretical model into practice.

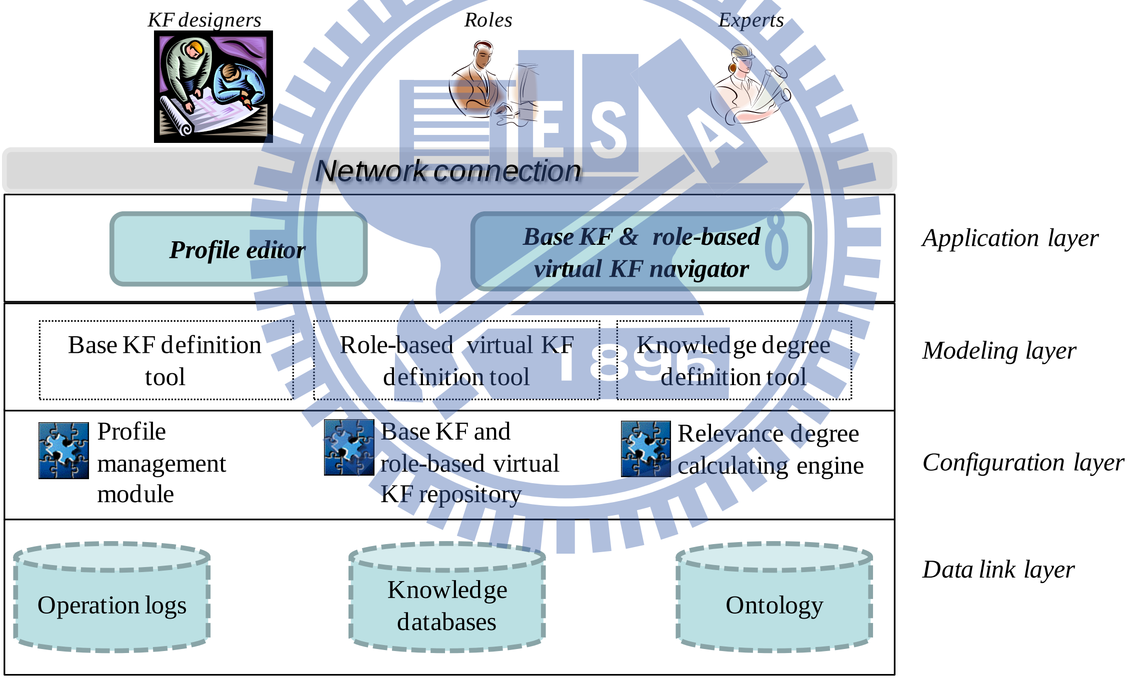


Figure 21. System architecture of the role-based KFV system.

Figure 21 depicts the system architecture to implement the role-based KFV system, which comprises four layers: data link layer, configuration layer, modeling layer and application layer.

Data link layer: This layer enables data links to other legacy systems to collect information from external data sources during the design time and run time. The operation logs preserve the history of roles' operations in workflow management systems. The knowledge databases store codified knowledge which is labeled by the knowledge concepts of domain ontology. Knowledge-based management systems manage these knowledge databases and provide interfaces for the role-based KFV system to access required codified knowledge. Domain ontology stores the pre-defined knowledge concepts and their hierarchical relationships for the purpose of representing knowledge-needs and facilitating knowledge concept abstraction.

Configuration layer: This layer comprises three parts: profile management module, base KF and role-based virtual KF repository as well as relevance degree calculating engine. KF designers and experts utilize the profile management module to collect essential information, such as role-operation relevance profiles, operation required knowledge concept profile and knowledge node profiles. The base KF and role-based virtual KF repository preserves model definitions and enactment instances. The role-knowledge node relevance degree calculating engine is responsible for obtaining the relevance degrees between roles and knowledge nodes.

Modeling layer: This layer includes three definition tools to define base KFs, role-based virtual KFs and roles' knowledge degrees. Roles use the knowledge degree definition tool to set *krdeg* to reflect role-operation knowledge requirement degrees based on their knowledge-needs. Meanwhile, KF designers work with experts to specify base KFs and corresponding role-based virtual KFs by base KF and role-based virtual KF definition tools, respectively.

Application layer: An integrated platform is built in this layer for the operations of KF designers, experts and roles. This layer mainly provides an interface for them to get visualization support and maintain profiles.

Figure 22 shows the activity diagram of the role-based KFV system. The actors in the activity diagram are KF designers, roles and experts. They perform these procedural activities and produce relevant material for generating role-base virtual KFs. The activity diagram is organized into three partitions to indicate the major responsible person of the activities of each partition. The rounded rectangles represent the activities which are performed by actors manually or executed by the tools of the role-based KFV system. And the rectangles show the material such as profiles, configurations or intermediate output that passed between activities. We omitted the detailed explanation of the activity diagram here because it is somewhat self-explanatory. The activity diagram is useful for system modeling to describe the control flow of the role-based KFV system, such as exploring the knowledge concept abstraction and knowledge node generation approaches, as well as the complex configuration evaluation and adjustment methods.

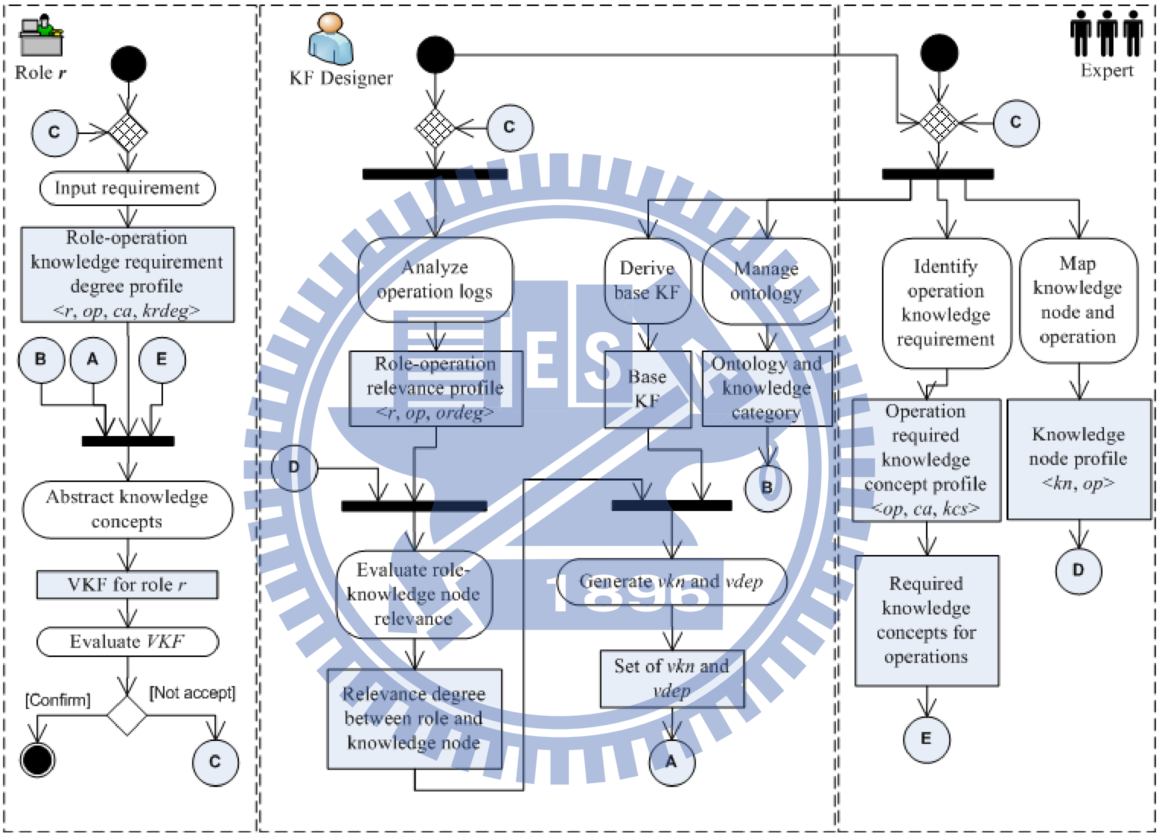


Figure 22. Activity diagram.

An approach of iterative evaluation and adjustment is adopted in the design to fine-tune profiles and configurations to improve the fitness level of virtual knowledge flows. At the bottom of the left partition in the activity diagram, role *r* should evaluate the fitness level of the generated virtual knowledge flow. If role *r* is satisfied with the virtual knowledge flow, the procedure stops. Otherwise, role *r* would reflect the discrepancies of current virtual knowledge flow to KF designers and/or experts. Then, they may adjust base KF, ontology, *krdeg*, or other configurations and estimation methodologies, and the regenerate virtual knowledge flows again until role *r* satisfies the result. For example, the iteration of *krdeg*’s adjustment is shown by the merging of a start point (●) and the label Ⓒ in the top of the left partition in Figure 22.

Protégé 4.0 software is utilized to build an ontology prototype, as shown in Figure 23, to represent a part of the knowledge concepts and their hierarchical relationships in mobile phone development domain. The process of ontology construction would include an evaluation and feedback mechanism to gradually improve ontology quality and result in a common understanding in organizations. The ontology prototype is appropriate for system designers to understand the concepts of ontologies in system design phase.

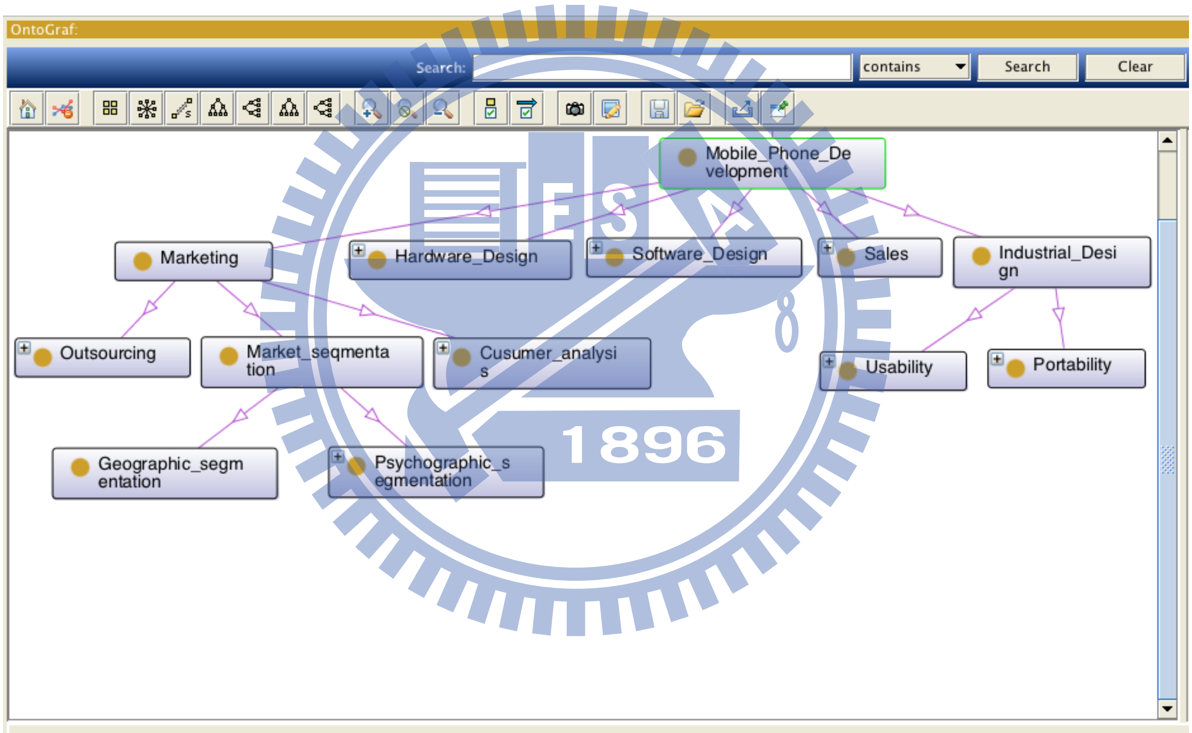


Figure 23. An ontology prototype.

5.5 Case illustration and analysis

This section uses the base knowledge flow (KF) of a mobile phone company as described in Section 4.5. The participants of the mobile phone development team work for different departments and play different roles in the team. For example, a sourcing planner role performs the logistics of parts outsourcing and a sourcing department manager role evaluates project performance and the sourcing planner role's productivity. The sourcing department manager role is responsible for communicating with the project manager about the project status and outsourcing strategy, as well as appraising the performance of the sourcing planner role. Therefore, KF designers can generate a role-based virtual KF from the base KF to represent the sourcing department manager role's knowledge-needs to support task execution.

The following discussion illustrates the process used to generate the role-based virtual KF. First, the base KF and the relevance degrees between the sourcing department manager role and the knowledge nodes are obtained by the approach described in Section 5.2. The base KF in Figure 24 includes knowledge nodes, k_0 to k_8. Each knowledge node contains multiple knowledge concepts and has distinct relevance degrees to r, herein; r stands for the sourcing department manager role.

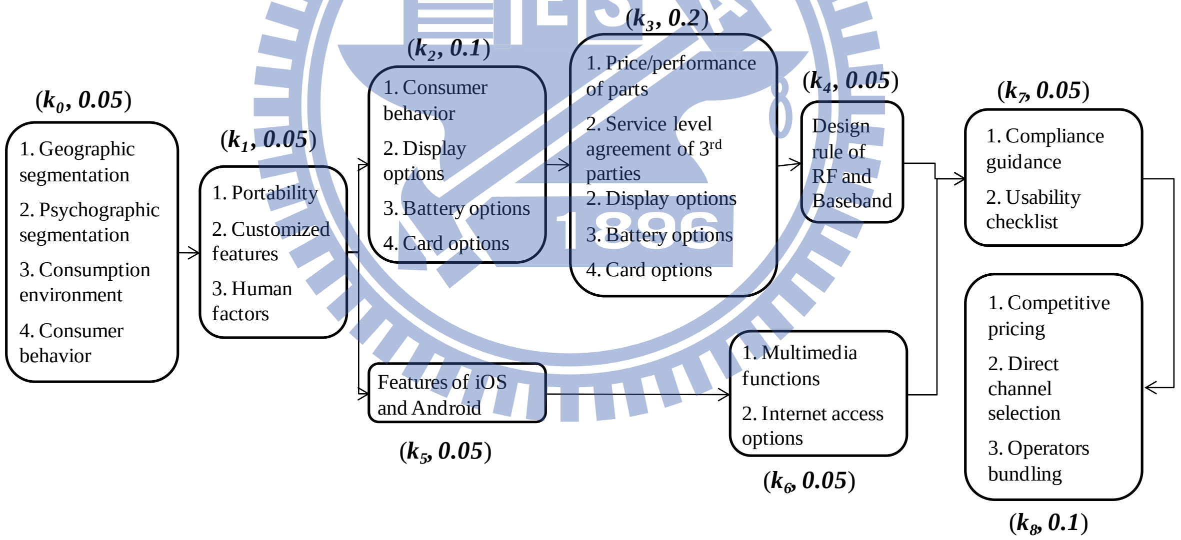


Figure 24. Relevance degrees between role r and base knowledge nodes.

Next, KF designers generate virtual KN based on a threshold (TH) 0.4. They select a base KN with the highest order, k_0, as a seed node to generate the first virtual knowledge node vkn_1. The adjacent knowledge node set $AKNS$ of k_0 is $\{k_1\}$. So, vkn_1 is considered a legal virtual knowledge node having two member knowledge nodes k_0, k_1. After checking F_{TRD} $(r, \{k_0, k_1\})$, KF designers can find $F_{TRD}(r, \{k_0, k_1\}) = f_{KRD}(r, k_0) + f_{KRD}(r, k_1) = 0.1 \leq TH$, which meet the threshold requirement. And vkn_1 also complies with the order-preserving property. So, vkn_1 is a legal VKN when it has two member KNs k_0, k_1. Because F_{TRD} (r, $\{k_0, k_1\}$) is not approximately close to TH, KF designers continue to evaluate $AKNS$ of $\{k_0, k_1\}$, which is $\{k_2, k_5\}$. According to the order-preserving property, we should add knowledge nodes k_2, k_3, k_4, k_5 and k_6 into vkn_1. However, F_{TRD} (r, $\{k_0 \ldots k_6\}$) = $\sum f_{KRD}$ (r, k_j) $\{j = 0 \ldots 6\}$ = 0.55 exceeds TH. Therefore, the first virtual knowledge node vkn_1 and its member knowledge nodes, k_0 and k_1, are determined. Repeating the iteration, the KNs are merged into four virtual knowledge nodes vkn_1, vkn_2, vkn_3 and vkn_4 to form a role-based virtual KF, as shown in Figure 25.

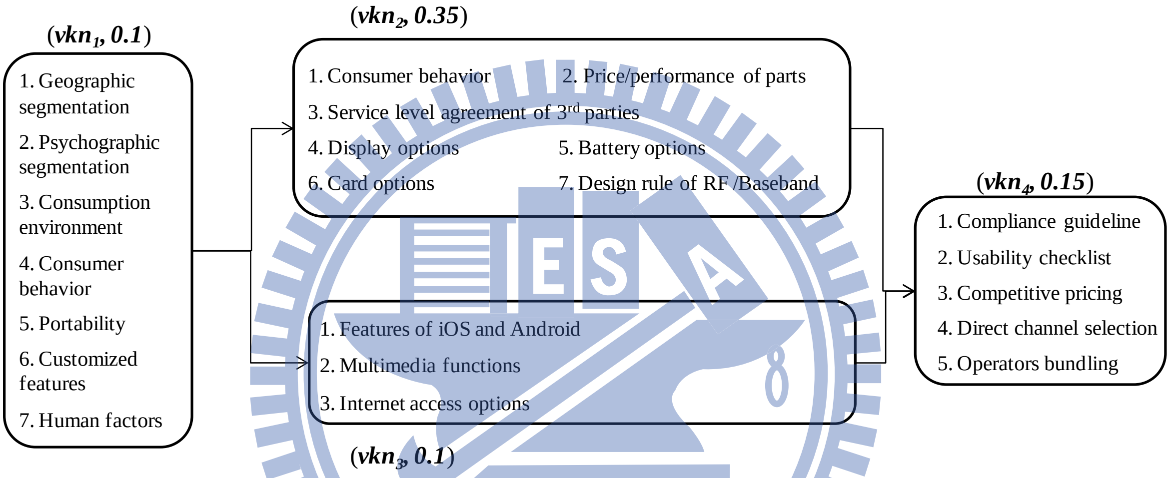


Figure 25. Virtual KNs and their relevance degrees when TH=0.4.

The following discussion takes virtual knowledge node vkn_2 as an example to illustrate the concept abstraction method. First, KF designers set r's knowledge-needs in terms of role-operation knowledge requirement degree *krdeg* by different knowledge categories. A partial domain ontology which includes different knowledge categories, such as: *marketing*, *industrial design*, *hardware design*, *software design*, *quality verification* and *sales*, is illustrated in Figure 26. It is to be noticed that the partial domain ontology is only used for concept explanation and case illustration here, instead of implementing a role-based KFV system in an organization. Ontology construction for organization use is a complex task and needs to further consider users' requirements, IT environments and the context of applications. This type of ontology is much more complete and complex than the partial domain ontology in Figure 26.

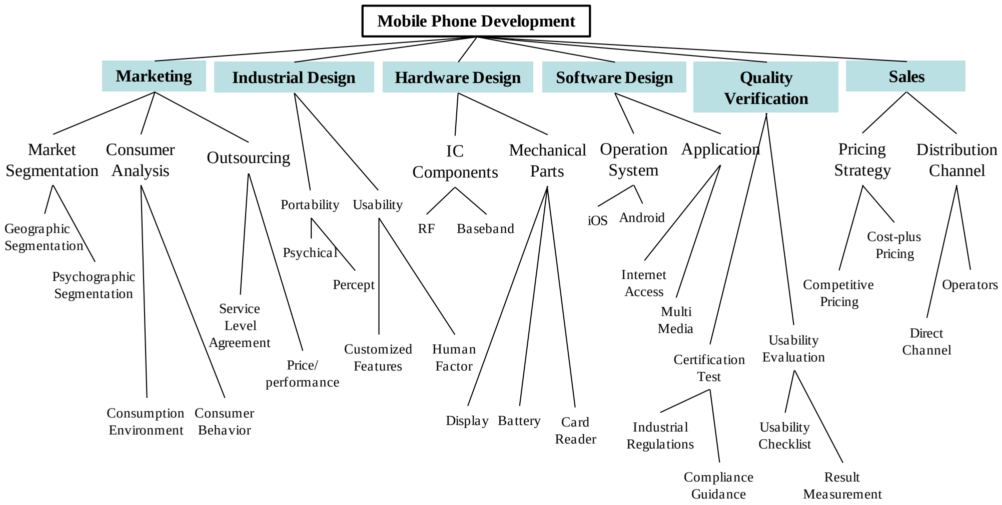


Figure 26. Partial knowledge categories of Marketing and Hardware Design.

Figure 27 shows the relevant information of the sourcing department manager role. According to the steps described in Section 5.3.2, the knowledge requirement degree $krdeg_{r,c}^{ca}$ represents the knowledge requirement degree of role *r* in regard to the knowledge concept *c* in the knowledge category *ca*. Hence, $krdeg_{r,c}^{ca}$ includes <*Marketing*, *consumer behavior*, *0.6*), <*Marketing*, *price/performance of parts*, *0.8*>, <*Marketing*, *service level agreement of 3rd parties*, *0.8*>, <*Hardware Design*, *battery options*, *0.6*>, <*Hardware Design*, *card options*, *0.6*>, <*Hardware Design*, *display options*, *1.0*>, <*Hardware Design*, *RF design rule*, *0.6*>, and <*Hardware Design*, *Baseband design rule*, *0.6*> through computing the operation required knowledge concepts and the role-operation knowledge requirement degrees *krdeg*.

Finally, the knowledge concepts are generalized by mapping knowledge degrees to adjusted concept levels. Consequently, the knowledge concepts of virtual knowledge node vkn_2 are *marketing*, *outsourcing*, *hardware design* and *display options*. The same method can apply to other virtual KNs. Figure 28 shows the final result of the role-based virtual KF.

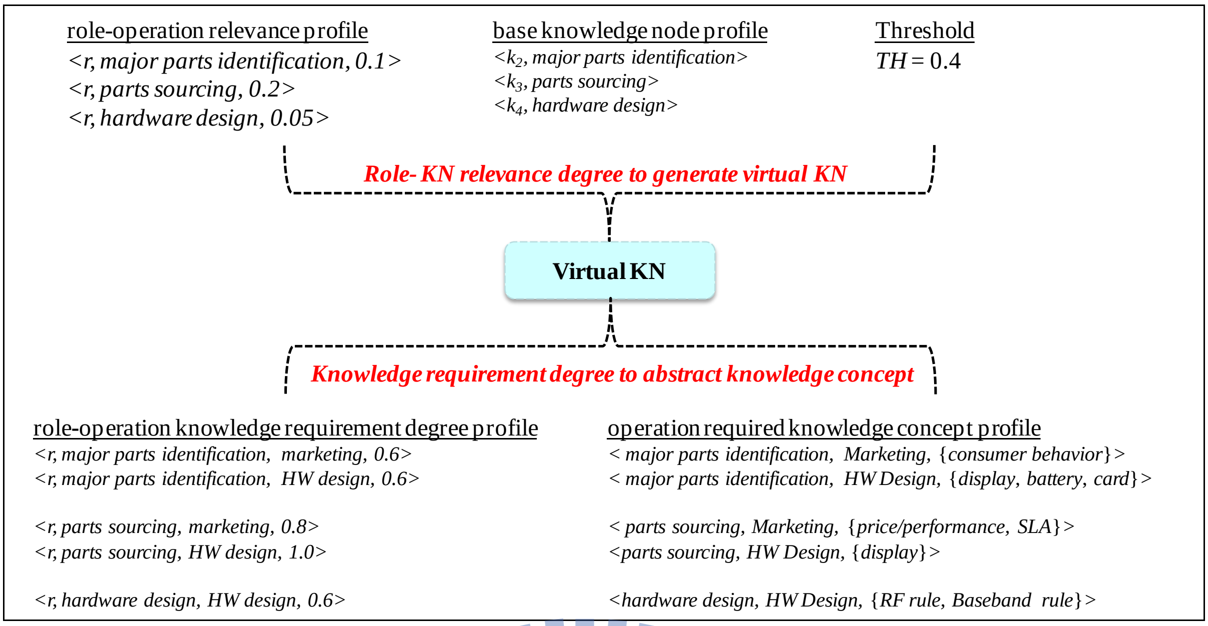


Figure 27. Information of sourcing department manager role.

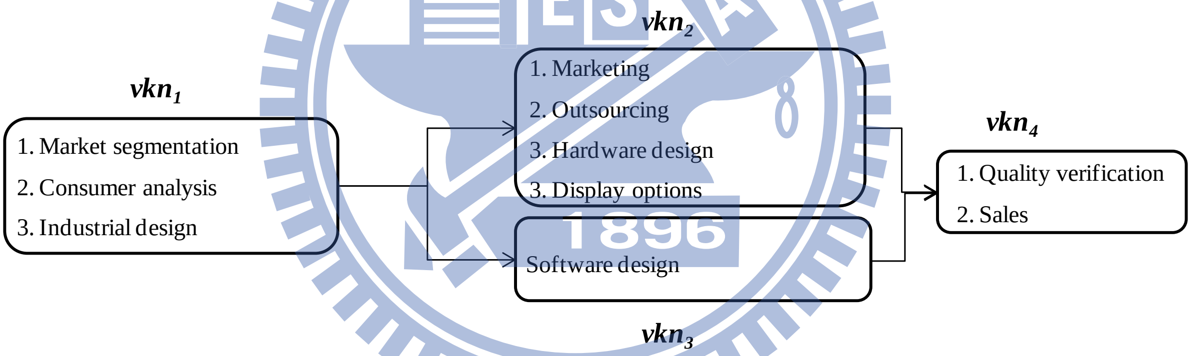


Figure 28. Role-based VKF for sourcing department manager role.

5.6 Discussion

This chapter presents the role-based approach for discovering role-based virtual KFs suitable for participating roles through applying the role-based framework. The r-KFV model is an extension of KFV model described in Chapter 4. The knowledge-needs of workers in a team may vary because they have different roles when they execute tasks. The role-based KFV model examines the worker's knowledge-needs in terms of his/her role. Discovering role-based virtual KFs involves two major steps: identifying role-based virtual knowledge nodes (virtual KNs) and deriving the knowledge concepts of virtual KNs. A role-based virtual KF comprises a set of virtual KNs, which are generated from the base KNs according to their relevance to the role. The relevance of a base KN to a given role can be specified by the KF designers or derived by analyzing the role's knowledge referencing behavior in document access logs. The relevance degree indicates the importance of a base KN to the role. Once the relevance of the aggregated base KNs reaches a certain threshold, a virtual KN is identified for the role. Then, KF designers can derive the knowledge concepts of the virtual KN based on the role's knowledge requirement degrees. The concept abstraction approach adjusts (generalizes) the knowledge concepts of virtual KNs to the appropriate concept levels based on the operation required knowledge concept profile and the knowledge requirement degrees of the role for the operations.

The proposed system architecture and activity diagram can be the base to conduct detailed system design for building functional specifications as well as for implementation. It is notable that a role-based KFV system tends to evolve in a longitudinal process. The back-and-forth between system evaluation and parameter adjustment is necessary for obtaining a well-run system; it will not be trivial work and needs lots of time and effort.

The r-KFV model brings practical benefits as follows: (1) the role-based virtual knowledge flows show roles with a full picture of knowledge-needs by presenting corresponding knowledge concepts with proper concept levels; (2) workers can describe their knowledge-needs precisely and gain a consensus quickly in teams by common domain ontology and role-operation knowledge requirement degree profiles; (3) KF designers can avoid complex and time-consuming tasks of estimating parameters for each different role; (4) the iterative evaluation and adjustment approach can fine-tune the fitness level of virtual knowledge flows to increase roles' satisfaction; and (5) the proposed model facilitates organizational knowledge support platforms to help teams improve their communication quality and increase members' productivity.

Chapter 6 Conclusions

6.1 Summary

This dissertation proposes the novel concepts of knowledge flow abstraction: knowledge-flow view model and virtual knowledge flows. The knowledge-flow view model enhances the conventional knowledge flow models to improve the effectiveness of knowledge sharing and knowledge support in organizations. In addition, the virtual knowledge flows illustrate teammates' different knowledge-needs precisely and facilitate collaborative knowledge provision in teamwork environments from task function and role perspectives.

In knowledge-intensive working environments, workers need task-relevant knowledge and documents to support task performance. To meet these requirements, many organizations have built knowledge support platforms that allow workers to preserve, share and reuse task-relevant knowledge. The value of knowledge support thus pertains to the importance of realizing knowledge management and promoting business intelligence in knowledge-based organizations.

Knowledge flow models have been proposed as an effective tool for building knowledge support platforms recently. Knowledge flows represent the flows of an individual's or group members' knowledge-needs and the referencing sequence of documents in conducting business operations and/or research activities. Through knowledge flows, organizations can facilitate knowledge support mechanism by providing knowledge to fulfill workers' knowledge-needs. However, the conventional KF models only provide single view to teammates, without considering participants different knowledge-needs according to their task functions and roles; this leads to decrease the efficiency of knowledge sharing in organizations. To satisfy team workers with different knowledge-needs, this work proposes the BKF model to illustrate knowledge-needs precisely, and the essential KFV model and r-KFV model, which are capable of generating multiple virtual knowledge flows.

The KFV model and r-KFV model construct virtual knowledge flows to serve workers' knowledge-needs. The knowledge-needs are arising from the gap between workers' knowledge and the necessary requirements of tasks, which including the fine-grand knowledge which workers need during performing jobs and the coarse-grand knowledge for team communication. To fulfill such knowledge-needs, virtual knowledge flows are derived from a base knowledge flow and provide abstracted knowledge.

According to the task functions, the KFV model builds virtual KFs based on concealing criteria and abstracts base KNs in a base KF to generate corresponding virtual KNs through an order-preserving approach and a knowledge concept generalization mechanism. In addition, the r-KFV model analyzes the levels of knowledge required by workers based on their roles, and develops role-based knowledge flow abstraction methods that generate virtual knowledge nodes to provide the appropriate level of knowledge for each role.

The virtual knowledge flows not only fulfill workers' different knowledge-needs but also facilitate knowledge support in teamwork. The concept of knowledge-flow view advances the conceptual applicability of knowledge flow research to cooperative knowledge support environments and helps researchers to obtain a clear view of knowledge flow research. It also improves the effectiveness of knowledge sharing and knowledge support in organizations.

6.2 Limitations and future work

One limitation of this work is the lack of a rigorous evaluation of the KFV models' practical benefits. Because this study constitutes fundamental knowledge flow research, it aimed to generate virtual knowledge flows and extend knowledge flow research to cooperative teams by establishing KFV models with novel methodology. The essential KFV model and r-KFV model could be the core for building KFV systems based on the theoretical contributions achieved in this work. Future work will build a KFV system to realize the practical benefits of the proposed models and its related methodologies. An empirical study will also be conducted to quantify user satisfaction, knowledge cooperation level and teamwork effectiveness by questionnaires or other measurement tools. The KFV system would consider the applicability in job enrichment of workers. In addition, KF designers can also utilize bottom-up approaches for collecting knowledge-needs from end-users while designing the KFV systems.

Another limitation of this study is that it does not consider how to integrate a workflow model. Organizations often adopt workflow models to manage the information in the business processes for task execution and team cooperation. This issue will be addressed in the future by designing a process-oriented KFV model to synthesize the information provided by workflow models. This process-oriented KFV model integrating workflow model and KFV models to coordinate and formulate the interactions between work flows and knowledge-flow views to facilitate knowledge dissemination in knowledge-based organizations. In addition, the abstraction methods of loop structure and other structures in the network of knowledge flow should be improved in the process-oriented KFV model to handle complicated workflow and knowledge flow applications.

Lastly, it is possible to obtain knowledge flows using other methods than the methods indicated in this paper. For example, knowledge flows might be derived from planning models generated by a knowledge-based planning methodology. In this way, the captured knowledge and derived constraints in the planning models would be the knowledge concepts of knowledge nodes. The dependency of planned tasks would establish the relationships between the knowledge nodes, forming a knowledge flow. Determining how to utilize such a knowledge-based planning methodology to generate knowledge flows systematically is another direction for future research.

References

[1] A. Abecker, A. Bernardi, K. Hinkelmann, O. Kuhn, and M. Sintek, "Context-aware, proactive delivery of task-specific information: the knowmore project", *Information Systems Frontiers*, 2 (3), pp. 253-276, 2000.

[2] A. Abecker, A. Bernardi, H. Maus, M. Sintek, and C. Wenzel, "Information supply for business processes: coupling workflow with document analysis and information retrieval", *Knowledge-Based Systems*, 13 (5), pp. 271-284, 2000.

[3] Y. Afacan and H. Demirkan, "An ontology-based universal design knowledge support system", *Knowledge-Based Systems*, 24 (4), pp. 530-541, 2011.

[4] H. J. Ahn, H. J. Lee, K. Cho, and S. J. Park, "Utilizing knowledge context in virtual collaborative work", *Decision Support Systems*, 39 (4), pp. 563-582, 2005.

[5] A. Anjewierden, R. de Hoog, R. Brussee, and L. Efimova, "Detecting knowledge flows in weblogs" in *Proceedings of the Thirteenth International Conference on Conceptual Structures*, pp. 1-12, Atlanta, 2005.

[6] S. Biundo, R. Aylett, M. Beetz, D. Borrajo, A. Cesta, T. Grant, L. McCluskey, A. Milani, and G. Verfaille, "Technological roadmap on AI planning and scheduling," PLANET, 2003.

[7] C. Chandra and A. Tumanyan, "Organization and problem ontology for supply chain information support system", *Data & Knowledge Engineering*, 61 (2), pp. 263-280, 2007.

[8] I. Chebbi, S. Dustdar, and S. Tata, "The view-based approach to dynamic inter-organizational workflow cooperation", *Data & Knowledge Engineering*, 56 (2), pp. 139-173, 2006.

[9] H. H. Chen and C. Pang, "Organizational forms for knowledge management in photovoltaic solar energy industry", *Knowledge-Based Systems*, 23 (8), pp. 924-933, 2010.

[10] Y.-J. Chen, "Development of a method for ontology-based empirical knowledge representation and reasoning", *Decision Support Systems*, 50 (1), pp. 1-20, 2010.

[11] S. Y. Choi, Y. S. Kang, and H. Lee, "The effects of socio-technical enablers on knowledge sharing: an exploratory examination", *Journal of Information Science*, 34 (5), pp. 742-754, 2008.

[12] H. K. H. Chow, K. L. Choy, and W. B. Lee, "A strategic knowledge-based planning system for freight forwarding industry", *Expert Systems with Applications*, 33 (4), pp. 936-954, 2007.

[13] D. T. Croasdell, M. Jennex, Z. Yu, T. Christianson, M. Chakradeo, and W. Makdum, "A meta-analysis of methodologies for research in knowledge management, organizational learning and organizational memory: five years at HICSS" in *Proceedings of the 36th Annual Hawaii International Conference on System Sciences*, 2003.

[14] P. F. Drucker, "The coming of the new organization", *Harvard Business Review*, 66 (1), pp. 45-53, 1988.

[15] T. C. Du, F. Li, and I. King, "Managing knowledge on the Web – Extracting ontology from HTML Web", *Decision Support Systems*, 47 (4), pp. 319-331, 2009.

[16] A. C. Edmondson and I. M. Nembhard, "Product development and learning in project teams: the challenges are the benefits", *Journal of Product Innovation Management*, 26 (2), pp. 123-138, 2009.

[17] R. Eshuis and P. Grefen, "Constructing customized process views", *Data & Knowledge Engineering*, 64 (2), pp. 419-438, 2008.

[18] T. R. Gruber, "A translation approach to portable ontology specifications", *Knowledge Acquisition*, 5 (2), pp. 199-220, 1993.

[19] Z. Guo and J. Sheffield, "A paradigmatic and methodological examination of knowledge management research: 2000 to 2004", *Decision Support Systems*, 44 (3), pp. 673-688, 2008.

[20] M. Havey, *Essential business process modeling*: O'Reilly Media, Inc., 2005.

[21] J. J. Huang, "The evolutionary perspective of knowledge creation - a mathematical representation", *Knowledge-Based Systems*, 22 (6), pp. 430-438, 2009.

[22] R. S. Huckman, B. R. Staats, and D. M. Upton, "Team familiarity, role experience, and performance: evidence from Indian software services", *Management Science*, 55 (1), pp. 85-100, 2009.

[23] Y. G. Ji, J. H. Park, C. Lee, and M. H. Yun, "A usability checklist for the usability evaluation of mobile phone user interface", *International Journal of Human-Computer Interaction*, 20 (3), pp. 207-231, 2006.

[24] J.-Y. Jung, K. Kim, D. Shin, and J. Park, "FlowWiki: A wiki based platform for ad hoc collaborative workflows", *Knowledge-Based Systems*, 24 (1), pp. 154-165, 2011.

[25] S. Kim, H. Hwang, and E. Suh, "A process-based approach to knowledge-flow analysis: a case study of a manufacturing firm", *Knowledge and Process Management*, 10 (4), pp. 260-276, 2003.

[26] J. Kingston and A. Macintosh, "Knowledge management through multi-perspective modelling: representing and distributing organizational memory", *Knowledge-Based Systems*, 13 (2-3), pp. 121-131, 2000.

[27] M. M. Kwan and P. Balasubramanian, "KnowledgeScope: managing knowledge in context", *Decision Support Systems*, 35 (4), pp. 467-486, 2003.

[28] C.-H. Lai and D.-R. Liu, "Integrating knowledge flow mining and collaborative filtering to support document recommendation", *Journal of Systems and Software*, 82 (12), pp. 2023-2037, 2009.

[29] S.-T. Li and H.-C. Hsieh, "Managing operation knowledge for the metal industry", *Journal of Universal Computer Science*, 9 (6), pp. 472-480, 2003.

[30] C.-W. Lin, D.-R. Liu, and H.-F. Chen, "Facilitating cooperative knowledge support by role-based knowledge-flow views" in *Proceedings of International Conference on Information and Knowledge Engineering*, Singapore, 2011.

[31] C. Lin, B. Tan, and S. Chang, "An exploratory model of knowledge flow barriers within healthcare organizations", *Information & Management*, 45 (5), pp. 331-339, 2008.

[32] H.-F. Lin, "A stage model of knowledge management: an empirical investigation of process and effectiveness", *Journal of Information Science*, 33 (6), pp. 643-659, 2007.

[33] D.-R. Liu and C.-H. Lai, "Mining group-based knowledge flows for sharing task knowledge", *Decision Support Systems*, 50 (2), pp. 370-386, 2011.

[34] D.-R. Liu and C.-W. Lin, "Modeling the knowledge-flow view for collaborative knowledge support", *Knowledge-Based Systems*, 31 pp. 41-54, 2012.

[35] D.-R. Liu, C.-W. Lin, and H.-F. Chen, "Discovering role-based virtual knowledge flows for organizational knowledge support," Institution of Information Management, National Chiao Tung University, Working paper, 2012.

[36] D.-R. Liu and M. Shen, "Workflow modeling for virtual processes: an order-preserving process-view approach", *Information Systems*, 28 (6), pp. 505-532, 2003.

[37] D.-R. Liu and M. Shen, "Business-to-business workflow interoperation based on process-views", *Decision Support Systems*, 38 (3), pp. 399-419, 2004.

[38] D.-R. Liu and I.-C. Wu, "Collaborative relevance assessment for task-based knowledge support", *Decision Support Systems*, 44 (2), pp. 524-543, 2008.

[39] P. Liu, B. Raahemi, and M. Benyoucef, "Knowledge sharing in dynamic virtual enterprises: A socio-technological perspective", *Knowledge-Based Systems*, 24 (3), pp. 427-443, 2011.

[40] X. Luo, Q. Hu, W. Xu, and Z. Yu, "Discovery of textual knowledge flow based on the management of knowledge maps", *Concurrency and Computation: Practice and Experience*, 20 (15), pp. 1791-1806, 2008.

[41] C. Marinica and F. Guillet, "Knowledge-based interactive postmining of association rules using ontologies", *IEEE Transactions on Knowledge and Data Engineering*, 22 (6), pp. 784-797, 2010.

[42] M. E. Nissen, "An extended model of knowledge-flow dynamics", *Communications of the Association for Information Systems*, 8 pp. 251-266, 2002.

[43] V. T. Nunes, F. M. Santoro, and M. R. S. Borges, "A context-based model for knowledge management embodied in work processes", *Information Sciences*, 179 (15), pp. 2538-2554, 2009.

[44] A. Öhgren and K. Sandkuhl, "Towards a methodology for ontology development in small and medium-sized enterprises" in *IADIS Conference on Applied Computing*, Algarve, Portugal, 2005.

[45] Á. E. Prieto and A. Lozano-Tello, "Use of ontologies as representation support of

workflows oriented to administrative management", *Journal of Network and Systems Management*, 17 (3), pp. 309-325, 2009.

[46] M. D. R-Moreno, D. Borrajo, A. Cesta, and A. Oddi, "Integrating planning and scheduling in workflow domains", *Expert Systems with Applications*, 33 (2), pp. 389-406, 2007.

[47] Y. Ren, K. M. Carley, and L. Argote, "The contingent effects of transactive memory: when is it more beneficial to know what others know?", *Management Science*, 52 (5), pp. 671-682, 2006.

[48] A. Riege, "Three-dozen knowledge-sharing barriers managers must consider", *Journal of Knowledge Management*, 9 (3), pp. 18-35, 2005.

[49] O. M. Rodríguez-Elias, A. I. Martínez-García, A. Vizcaíno, J. Favela, and M. Piattini, "A framework to analyze information systems as knowledge flow facilitators", *Information and Software Technology*, 50 (6), pp. 481-498, 2008.

[50] V. Sambamurthy and M. Subramani, "Special issue on information technologies and knowledge management", *MIS Quarterly*, 29 (1), pp. 1-7, 2005.

[51] I. Savvas and N. Bassiliades, "A process-oriented ontology-based knowledge management system for facilitating operational procedures in public administration", *Expert Systems with Applications*, 36 (3), pp. 4467-4478, 2009.

[52] K. A. Schulz and M. E. Orlowska, "Facilitating cross-organisational workflows with a workflow view approach", *Data & Knowledge Engineering*, 51 (1), pp. 109-147, 2004.

[53] M. Shen and D.-R. Liu, "Discovering role-relevant process-views for disseminating process knowledge", *Expert Systems with Applications*, 26 (3), pp. 301-310, 2004.

[54] V. Sugumaran and V. C. Storey, "Ontologies for conceptual modeling: their creation, use, and management", *Data & Knowledge Engineering*, 42 (3), pp. 251-271, 2002.

[55] Z. S. Syed, T. Finin, and A. J, "Wikipedia as an ontology for describing documents" in *Proceedings of the Second International Conference on Weblogs and Social Media*, Seattle, Washington, 2008.

[56] M. Uschold and M. Gruninger, "Ontologies: Principles, methods and applications", *Knowledge Engineering Review*, 11 (2), pp. 93-136, 1996.

[57] M. Uschold and M. King, "Towards a methodology for building ontologies" in *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, 1995.

[58] W.-T. Wang and S. Belardo, "The role of knowledge management in achieving effective crisis management: a case study", *Journal of Information Science*, 35 (6), pp. 635-659, 2009.

[59] Y.-F. Wen, "An effectiveness measurement model for knowledge management", *Knowledge-Based Systems*, 22 (5), pp. 363-367, 2009.

[60] S.-S. Weng and H.-L. Chang, "Using ontology network analysis for research document recommendation", *Expert Systems with Applications*, 34 (3), pp. 1857-1869,

2008.

[61] M. Weske, *Business process management: concepts, languages, architectures*: Springer, 2007.

[62] I.-C. Wu, D.-R. Liu, and P.-C. Chang, "Toward incorporating a task-stage identification technique into the long-term document support process", *Information Processing & Management*, 44 (5), pp. 1649-1672, 2008.

[63] I.-C. Wu, D.-R. Liu, and P.-C. Chang, "Learning dynamic information needs: a collaborative topic variation inspection approach", *Journal of the American Society for Information Science and Technology*, 60 (12), pp. 2430-2451, 2009.

[64] Y. Xu and A. Bernard, "Quantifying the value of knowledge within the context of product development", *Knowledge-Based Systems*, 24 (1), pp. 166-175, 2011.

[65] C. C. Yang, J. Lin, and C.-P. Wei, "Retaining knowledge for document management: category-tree integration by exploiting category relationships and hierarchical structures", *Journal of the American Society for Information Science and Technology*, 61 (7), pp. 1313-1331, 2010.

[66] Z. Zhang, Z. Yang, and Q. Liu, "Modeling knowledge flow using Petri Net" in *IEEE International Symposium on Knowledge Acquisition and Modeling Workshop*, pp. 142-146, China, 2008.

[67] J. L. Zhao, H. H. Bi, H. Chen, D. D. Zeng, C. Lin, and M. Chau, "Process-driven collaboration support for intra-agency crime analysis", *Decision Support Systems*, 41 (3), pp. 616-633, 2006.

[68] W. Zhao and W. Dai, "Integrated modeling of business processes and knowledge flow based on RAD" in *IEEE International Symposium on Knowledge Acquisition and Modeling Workshop*, pp. 49-53, China, 2008.

[69]H. Zhuge, "Knowledge flow management for distributed team software development", *Knowledge-Based Systems*, 15 (8), pp. 465-471, 2002.

[70] H. Zhuge, "A knowledge flow model for peer-to-peer team knowledge sharing and management", *Expert Systems with Applications*, 23 (1), pp. 23-30, 2002.

[71] H. Zhuge, "Discovery of knowledge flow in science", *Communications of the ACM*, 49 (5), pp. 101-107, 2006.

[72] H. Zhuge, "Knowledge flow network planning and simulation", *Decision Support Systems*, 42 (2), pp. 571-592, 2006.

[73] H. Zhuge and W. Guo, "Virtual knowledge service market - for effective knowledge flow within knowledge grid", *Journal of Systems and Software*, 80 (11), pp. 1833-1842, 2007.