

Chapter V

Application and Scenarios

V.1. Introduction

In this chapter we aim at developing the Knowledge Management System for the ceramic cluster regarding the requirements and specification which are obtained from the processes in chapter 4. The organization of this chapter will follow the provided services in the system architecture: collaboration, knowledge, and presentation. In the first part of this chapter, the complete system architecture will be presented in order to provide the integrated view of the system in terms of services and inter-connection between each service. After that, the system architecture will be broken down into class level of each function. Particular functions that support the service will be depicted by database diagram, class diagram, and also the user interface. Finally, the application of the KMS will be demonstrated by using different scenarios of ceramic cluster activities.

Lastly, we demonstrate three scenarios which present the interaction between KMS and the activity of the cluster. The first scenario is based on collaborative activity of the members. It demonstrates how the cluster committee organizes a meeting via the virtual conference module. It also proves that three modes of communication: individual, group, and mass communication were supported in the KMS. The second scenario focuses on the knowledge management process. It shows how experts in the cluster are able to represent and share their knowledge via the Collaborative-Knowledge Card module. Then, the inference engine was used for retrieving the knowledge from the system semantically. The last scenario presents the integration of collaboration and knowledge service together. The result of the assimilation allows knowledge users to retrieve knowledge from KMS anywhere and anytime via their mobile device.

V.2. The proposed KMS architecture

In this section, the system architecture of the KMS for the ceramic cluster will be illustrated in order to visualize the subsystem, functions, services, tools, and interface between each element in the system. The proposed system architecture in the design level of the framework will be ameliorated in order to present functionality of the service. The functional view of the system is illustrated in figure V.1. Furthermore, the connection aspect of system architecture was designed in order to demonstrate the interface between the modules in the system. The system interface diagram shows the principle modules of the system with required protocols for interfacing with each specific module. The proposed system architecture was designed on the web services concept which supports the interoperability between machine-to-machine interactions over a network. This concept is quite useful to our knowledge system in order to retrieve the information from another web service server and present it to the cluster member. For example, the collaborative calendar service in the knowledge system exchanges information with Yahoo's server by using iCalendar standard (RFC 2445). Another advantage of the web service is that the cluster members do not need to invest on new information system infrastructure.

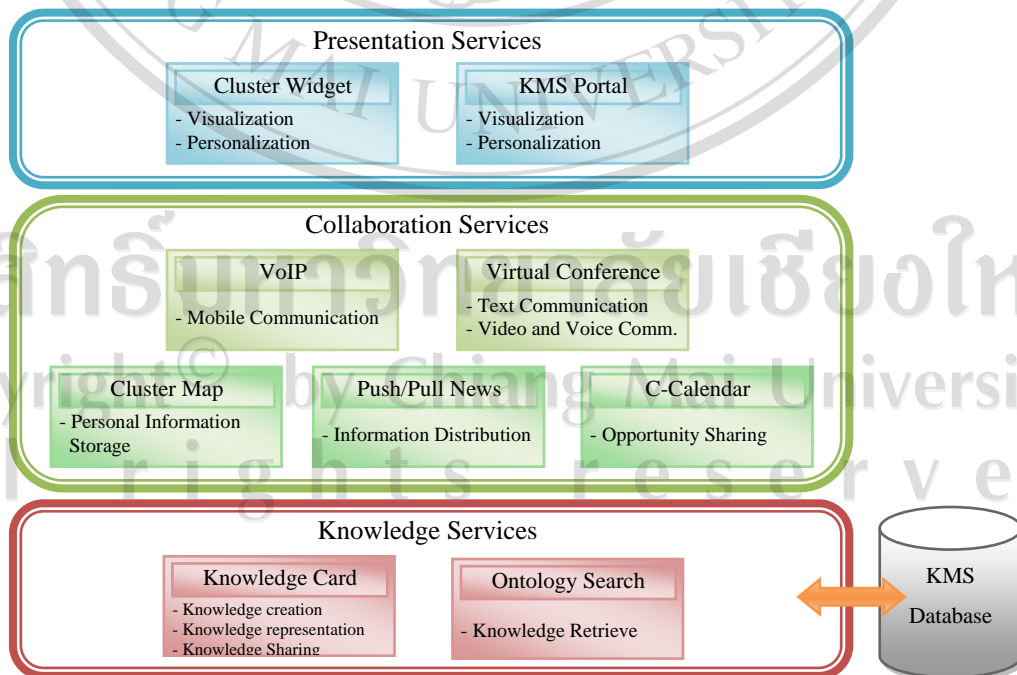


Figure V.1: KMS system architecture for the ceramic cluster

The architecture above shows the proposed functions which support the activities in each service. The knowledge service level is fundamental to the knowledge system. It aims at supporting the knowledge activities of the ceramic cluster: knowledge creation, representation, sharing, and retrieval. This service was supported by two modules: knowledge card module and ontology search module. The second level is a collaboration service which focuses on improving the collaboration of the ceramic cluster. This service is divided into two levels: communication and collaboration. The communication level (light green) comprises the VoIP module and virtual conference module. These modules aim at enhancing the quality and facilitating communication among the cluster members. The collaboration level (dark green) includes cluster map, push/pull news and collaborative calendar. These modules focus on the collaborative tools which support the critical activity of the ceramic cluster. The details of each tool will be described in the next section. The top level introduces the presentation services. The objective of this level is delivering information and knowledge from the knowledge system to the knowledge users at the right place, time, format and quality. Moreover, it allows the users to personalize and visualize the knowledge to suit their preferences. The presentation service comprises two modules: the KMS portal and the cluster widget. In order to gain better comprehension about the system architecture is illustrated in figure V.2.

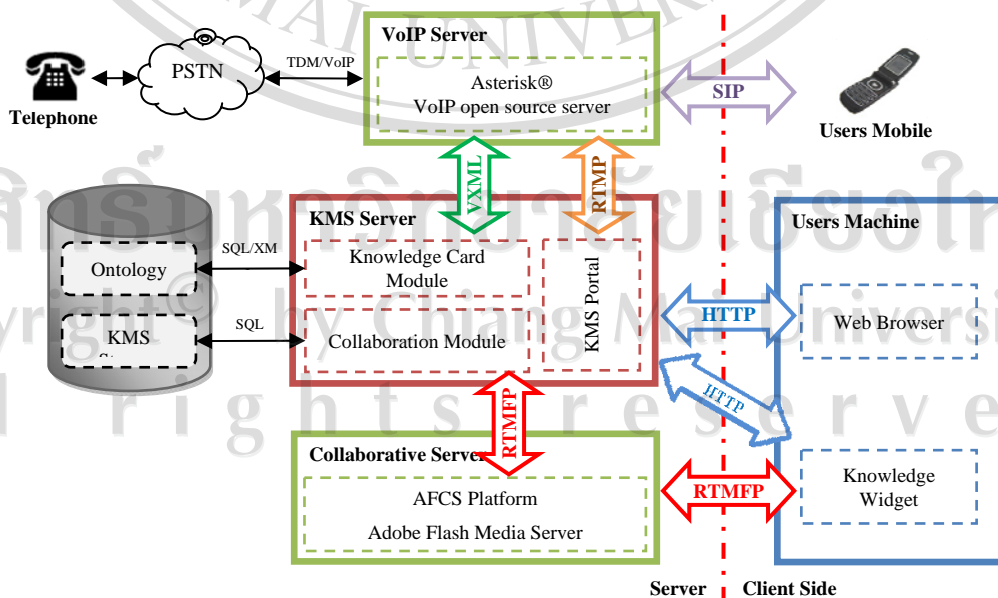


Figure V.2: The interface model of the system architecture

The interface model demonstrates the inter-connection between each module in our KMS. The left-hand side of vertical dashed line denotes the server side, while the right-hand side is the client machine. The fundamental of the system is composed of three types of server: KMS, collaborative, and VoIP.

Server Side

The KMS server is the centre of the system. It includes knowledge services, collaborative functions, and KMS portal of presentation services. The principal databases of the system are located within this server. MySQL database was selected for storing data of the system. The SQL (Structured Query Language) and XML (eXtensible Markup Language) are the major approaches which are used for retrieving data from the database. The collaboration module handles the information sharing of the cluster members. Moreover, it also takes care of the connection between users and collaborative server via the protocol RTMFP (Real Time Media Flow Protocol). Then, the knowledge card module was built to support knowledge activities of the cluster. The data structure of this module is based on XML format to ease exchanging between different platforms. The KMS Portal was developed on PHP 5 and AJAX (Asynchronous JavaScript and XML) technology which permits the cluster member to browse and access all the provided services on the KMS via client web browser via HTTP (HyperText Transfer Protocol). Furthermore, this module facilitates the access to the VoIP server by encapsulating HTTP within RTMP (Real Time Messaging Protocol). The protocol encapsulation will be explained in VoIP function in the next section.

The collaborative server is implemented with the intention of enhancing the quality of collaboration of the industry cluster. This server allows users to collaborate via virtual conferencing. This means that members are able to send/receive text, voice, and video to each other in real time. In order to achieve this goal, this study adopted the AFCS platform which is short for Adobe Flash Collaboration Service. The AFCS is a platform-as-a-service [AFCS 09] that allows systems to add real-time social capabilities to the Rich Internet Application (RIAs). The examples of available services on this platform are roster, chat, webcam, etc. The features and architecture of this platform will be depicted in the virtual conference section in this chapter.

Another service that aims to ameliorate the communication of the members is VoIP (Voice over Internet Protocol). This service enables members to make a phone call (deliver voice) over IP network without any additional cost. In our study, we have integrated Asterisk® which is an open source telephone engine that allows us to establish VoIP service in the KMS server. It also provides many benefits to cluster members and their businesses. A user could call other cluster members who are connecting with KMS server without any telephone costs. Furthermore, it can be used for calling customers in long distance with lower rate than landline telephone. From the diagram in figure V.2, it is implied that there are three methods to communicate with VoIP server. The common communication protocol for VoIP service is Session Initiation Protocol (SIP). SIP provides us an approach to transmit voice and video over the internet. It seems that more and more mobile phones support this protocol which provides an alternative way for users rather than GSM service providers. The second method is connecting from ordinary Public Switched Telephone Network (PSTN). This type of connection requires special hardware to interface with normal telephone lines. However, this connection allows users to make a call from VoIP phone to ordinary telephone sets. Lastly, the alternative way is connecting via the web service that provided in the KMS portal. Cluster members can access the VoIP service without installing any software in their machines. Therefore, the RTMP protocol encapsulation, mentioned in the previous paragraph, is required.

Client Side

On the client side, users have two ways to communicate with the KMS server. The first approach is accessing via web browser in their machine. While cluster members logged in to the KMS portal, KMS server will handle the connection to all provided services. The requested services will be encapsulated in HTTP and sent back to the user. An alternative approach is accessing via the knowledge widget. The knowledge widget is a small desktop application which was developed in our study. The widget provides the Graphic User Interface (GUI) look which allows cluster members to personalize their information, and also access all KMS services from one application without protocol encapsulation. The details of knowledge widget for industry cluster will be described in the widget section in this chapter.

V.3. Knowledge Management services

This level of the service is the fundamental of the knowledge system, which aims at facilitating knowledge management activities of the industry cluster (refer to our proposed system architecture; the main knowledge services that are supported in the knowledge system are knowledge creation, representation, sharing, and reusing). These activities are considered as the heart of the knowledge management. Actually, the most important objective of these services is to transfer the required knowledge from the experts to the knowledge users at the right place and on the right time. In order to improve these activities, the role of KMS and technology which support the activity were considered.

- *Knowledge representation* may be called by different names such as organizing, customizing or codifying knowledge. The objective of this process is representing the knowledge to suit the knowledge selection. In the past, representing the knowledge was done by writing, drawing or coding the knowledge into a codified format (e.g. procedure). Nowadays, thanks to the emergence of multimedia and information technology, knowledge can be represented in various forms and formats such as sound, picture, video, concept map, 3-D model, database, etc. These technologies make it possible for experts to articulate their tacit knowledge into explicit knowledge, which has been almost impossible in the past.
- *Knowledge sharing* (also known as knowledge transferring) is disseminating personal knowledge in the explicit form over the organization through a specific medium. This process is a core of the KMS which allows knowledge users in the industry cluster to acquire the required knowledge from the experts via the KMS.
- *Knowledge utilization* is integrating the knowledge from the KMS to the organization. In the knowledge engineering domain, this process refers to providing the right knowledge to the right person at the right time, right place and right format. Thus, knowledge storage and retrieval are considered as a key technology that supports the knowledge workers to complete their tasks.

In this study, the mobile service was proposed to decrease the limitations in terms of place and time. It allows the knowledge user to acquire the knowledge from the knowledge-base in KMS via mobile technology such as telephone, soft-phone or web services. The details of mobile technology will be explained in the following section.

In order to combine these services into our knowledge system, two major modules are integrated into the system: knowledge card [Buzon 03] and knowledge search module. The knowledge card module supports knowledge representation and the knowledge sharing activity of the cluster members. It allows the cluster's experts to express their knowledge in different modes such as text content, semantic map, audio and video stream, etc. Moreover, it also allows experts in the same Community of Practice (CoP) to jointly share knowledge on the same task. The explicated knowledge is stored in the knowledge system in the knowledge card format. The details of the knowledge card will be described in the next section. Then, the knowledge search module will be used to retrieve the knowledge from the knowledge system. This module allows knowledge workers to reuse the knowledge which has been accumulated within the knowledge system whenever they need to do so. Therefore, this section will focus on both modules. An overview of the components of knowledge services is illustrated by a UML diagram, as shown in figure V.3.

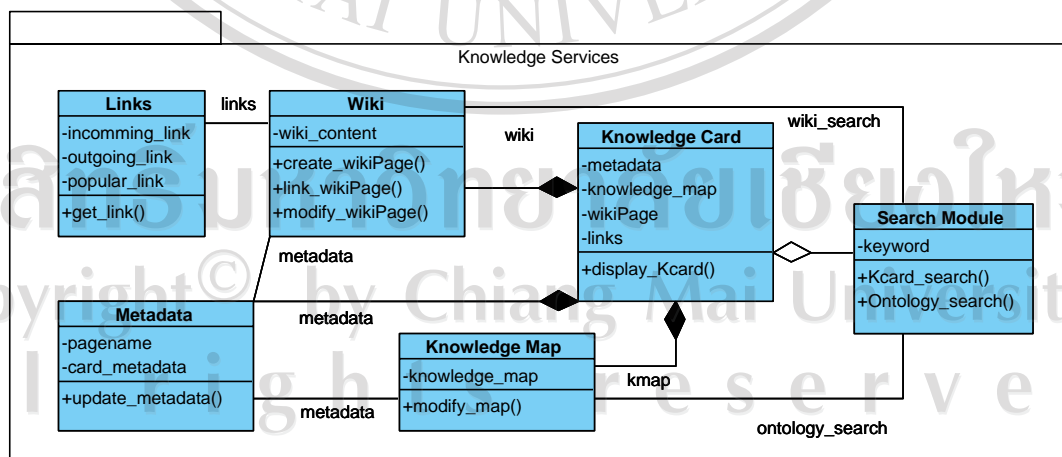


Figure V.3: The components of knowledge services

The diagram above implies that knowledge services are composed of two main modules. The first module is the knowledge card which contains 4 components: metadata, knowledge map, wiki, and links. The metadata component provides the card's name and metadata of the knowledge card. The knowledge map provides the knowledge model that was derived from the knowledge elicitation process. The wiki component allows experts to describe their knowledge model in descriptive approach. The links component shows the links of specified knowledge to the relative cards. The second module is the search module. This module allows the knowledge worker to browse through the knowledge card via two methods, wiki search and ontology search. The details of these search methods will be described in the knowledge search section.

V.3.1. Knowledge cards

In order to make use of the knowledge maps, the notion of knowledge card [Buzon 03] was adopted in our knowledge system as the medium of exchange. The knowledge card allows knowledge users to retrieve the knowledge map, information, and metadata of each task in a single card. A knowledge card represents one concept about a particular task and is linkable to other cards. Thus, this concept permits the experts in the cluster to create their knowledge card and attach to others on the same topic. Supplementing new knowledge to the system will improve the completeness of the knowledge for achieving the knowledge intensive task. In this part, we will depict the overview of the knowledge card and its structure. Then, the knowledge map which is the core of the knowledge card will be described. Finally, the system architecture of the knowledge card module from the points of view of data, class, and user interface will be presented.

V.3.1.1. Knowledge card's outline

The concept of knowledge card was adopted in this study in order to present the knowledge about a specific task in the form of a card. In a single card, all necessary information will be displayed to the reader. The information displayed in

the card is metadata, body of knowledge, description, and links. The elements in the knowledge card are structured as shown in figure V.4.

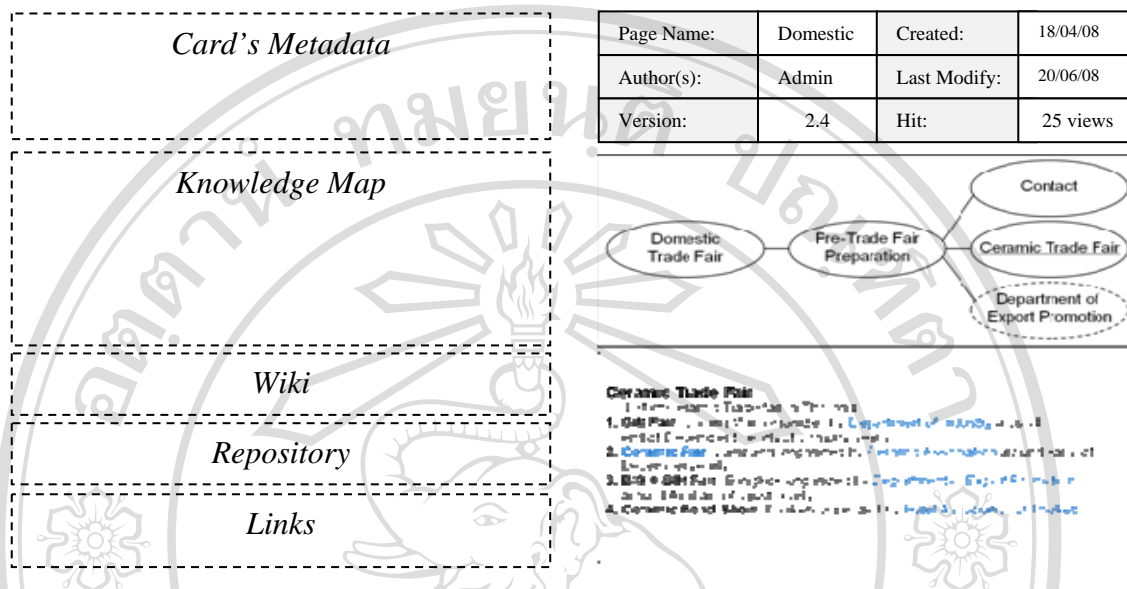


Figure V.4: An outline of the knowledge card

A knowledge card comprises five elements: card's metadata, knowledge map, wiki, repository and links. These elements help knowledge workers to comprehend and search for required knowledge through the KMS.

- Card's Metadata shows general information about the knowledge card such as author, version, last modified date, etc. This part notifies the reader about the popularity, version, and permission of the user on the specific card.
- Knowledge Map displays concept of a knowledge card in the form of a semantic map. This is the core of the knowledge card because it allows machine and human to browse over the cards. It also aims at representing the experts' knowledge into semantic map form in order to facilitate knowledge sharing, and reuse. The details of this will be discussed in the following part.
- Wiki module displays the collaborative knowledge base that is created and modified by experts in the same community of practice. This part allows users to share their knowledge which could not be represented by the knowledge map module.

- Repository displays a list of documents (files, databases, images, videos, etc.) that supports or is related to the knowledge card.
- Links part displays the incoming, outgoing, and popular links which are obtained from the metadata of knowledge cards.

With these elements, experts in the cluster are able to explicate their knowledge and information in various methods. They are able to manipulate (i.e. create and modify) the knowledge card concurrently via knowledge representation module in KMS. Knowledge cards are linked to each other by the inference engine and stored in the knowledge base for further modification or acquisition. The inference engine automatically links the domain knowledge with equivalent knowledge cards in the knowledge base.

The advantage of the knowledge card is that it conveys brief and concrete knowledge about a single task to the reader. A knowledge card signifies one task of the industry cluster. There are two methods to reuse the knowledge cards in the knowledge system. The first method is accessing via the knowledge representation module. Knowledge users should start browsing from the first card of the community of practice. This card will illustrate all possible tasks which are involved with particular practice. Then, users are able to browse from the general task to the more specific tasks via the knowledge map module. The second method is accessing the required knowledge card via the search module. This method allows knowledge users to precisely reuse their knowledge from the knowledge base. Moreover, it also facilitates knowledge retrieval from other applications. The details of the knowledge map component will be explained in the following part.

V.3.1.2. Knowledge map component

The knowledge map module is based on the concept of Task-Inference-Domain which was proposed in the CommonKADS methodology [Schreiber 99]. This module aims at representing expert's tacit knowledge into explicit knowledge in the format of knowledge model. The knowledge model comprises three elements (task, inference and domain knowledge), each capturing a related group of knowledge structures.

- *Task* knowledge describes what goal(s) an application pursues, and how these goals can be realized through the decomposition into subtasks and inference.
- *Inference* knowledge describes the basic inference steps required for making use of the domain knowledge. The inference level is also employed by the inference engine for retrieving required knowledge.
- *Domain* knowledge specifies the domain-specific knowledge and information focused on the particular task. The knowledge modeling allows experts in the industry cluster to present their knowledge in graphical form, also called the knowledge map.

Regarding the CommonKADS methodology, the knowledge map is notated as UML class diagram (without operation) which permits the experts and knowledge engineers to clarify and manipulate the structure of knowledge intensive task. Although the UML class diagram is suitable representation for the inference engine, it is not convenient for the knowledge user who is not in the computer domain to comprehend the knowledge map. Thus, this study adopted the concept of semantic concept map which is the integration of CommonKADS knowledge model and semantic map concept. So, the modeled knowledge is more comprehensible to the knowledge user and still readable to the machine. Another advantage is enabling the relationship between two concepts in both directions (i.e. forward and backward). The defined relationships help the user to read the knowledge map semantically and also allow the machine to do semantic search in the knowledge map. Figure V.5 shows an example of modeling experts' personal knowledge into semantic concept map format.

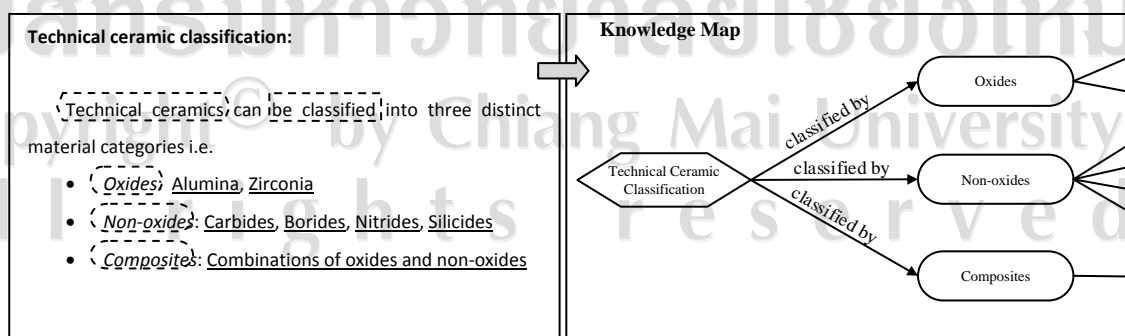


Figure V.5: Personal knowledge and the knowledge model

The figure above demonstrates an approach for representing experts' knowledge in the semantic map format by using the notion of Task-Inference-Domain. The knowledge about "technical ceramic classification" was presented on the left hand side as an example. During this process, the knowledge engineer employed the knowledge elicitation technique which was described in chapter 3 to extract the body of knowledge from the expert. Many knowledge elicitation methods such as interview, questionnaire, or focus group, were proposed for achieving this process. Then, the body of knowledge (i.e. task, inference, and domain knowledge) was notated and modeled into semantic map format as shown on the right hand side in the figure. The task knowledge was symbolized by the hexagonal shape. This type of knowledge was modeled as the parent node of the knowledge model. Consequently, the domain knowledge was symbolized by the oval shape and linked with their parent concept node. Finally, the relationships between task and domain knowledge were designated as the inference. These processes will be repeated until all domain knowledge about the specified task is modeled.

The semantic map makes it possible for the experts to explicate their knowledge in graphical form. The nature of a semantic map structure allows cluster experts to easily add new knowledge concepts onto other's knowledge. The shared knowledge is focused on a single task which is easy for the practice. The major advantage of a semantic map to the knowledge system is enabling the search module to acquire the required knowledge from the knowledge system. Each knowledge concept is linked to each other with semantic relationships, i.e. forward and backward. These relationships support humans to comprehend the interaction between the concepts, and also assist the machine to search for the required knowledge via the semantic search module. The forward relationship is visible in the knowledge card for reading by the human. In contrast, the backward relationship is concealed from the knowledge card, but employed by the machine for backward inference. The semantic map was stored in the format of XML, based on the topic map standard (ISO/IEC 13250:2003). Complying with this standard provides interoperability to the knowledge system for exchanging the knowledge with other systems. For a better understanding about the semantic concept map, figure V.6 shows a simple semantic

knowledge map with both forward and backward relationships and the map's structure in the XML format.

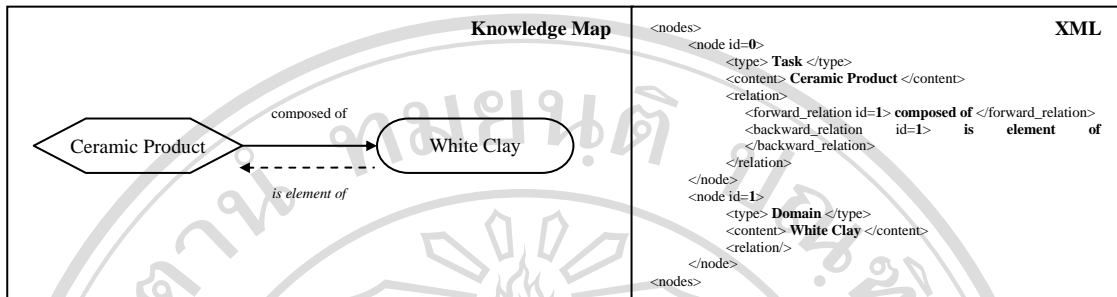


Figure V.6: Semantic concept map with XML code

The figure above shows the knowledge map compared to its XML form. The knowledge is map composed of two nodes, task and domain concept. Each node is identified by node id, as shown in the XML. The forward relationship is used for specifying the relationship from task concept (parent node) to the domain concept (child node). Conversely, the backward relationship is used for specifying the relationship from the child node back to the parent node. These relationships are employed by the search module in order to retrieve the knowledge from the knowledge base, which will be explained in the search module section.

In this study, there are two methods for creating the knowledge card, knowledge elicitation and collaborative knowledge sharing. Knowledge elicitation is utilizing the elicitation method in order to extract knowledge from the experts in the cluster. CommonKADS knowledge engineering methodology was adopted for converting experts' tacit knowledge into the knowledge model. The elicitation processes were described in the knowledge model section in chapter 3. Another method is allowing experts in the industry cluster to share their knowledge collaboratively. This method is comparable to a card sorting technique but in the virtual way. This means that the knowledge card system allows experts to add their knowledge (index card) to others' knowledge on a focused topic. Then, experts provide the relationship between their concept and relative concepts. Although the obtained knowledge card from this approach is not in well-structured form, it gives a possible method for the experts in the industry cluster to create and represent their knowledge with less effort. One advantage of the knowledge model is that it is easy to

manage the concept. Experts can share their ideas on the same topic by using less effort than for editing text or wiki content. However, experts are able to describe more ideas about particular knowledge in the knowledge-based wiki section, which will provide the complete information.

This part explained the methodology for enabling knowledge sharing for the industry cluster. The knowledge card was considered as the medium for transferring the knowledge from the experts to knowledge users. In the next section, knowledge card architecture will be used for explaining, from the global view to class view, the knowledge service level.

V.3.1.3. Knowledge card architecture

In order to get a better understanding about the knowledge card module, the architecture of knowledge card will be described in this section. The base engine of the knowledge card is wiki engine. In this study, we have employed PHPWiki [PHPWiki 08] as a core of these modules. It is an open source wiki engine which allows cluster members to collaboratively manipulate the content pages through an HTML form. Linking the wiki pages is done automatically on the server side; all pages are stored in the database. Due to this engine being developed on open source agreement, it allows us to modify the source code in order to add the knowledge card module over the wiki system. The database of the system was customized to suit our knowledge system. The database schema of the knowledge card is illustrated in figure V.7.

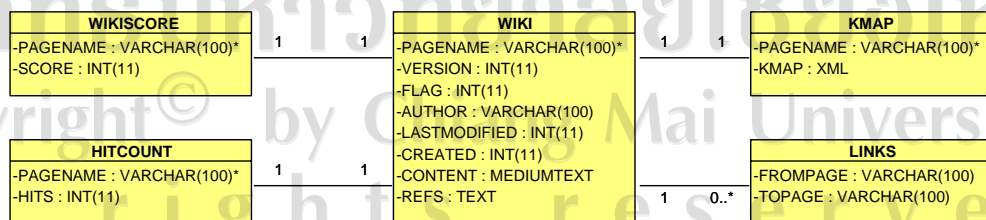


Figure V.7: Database schema of knowledge card system

The knowledge card module includes five main database tables: Wiki, WikiScore, HitCount, Kmap, and Links. The primary key of these tables is the page

name (or card name). Wiki table contains metadata and wiki content of the card. WikiScore and HitCount are used by the wiki engine to provide recommended card or popular card to the reader. Then, Kmap table contains knowledge map in XML format. Lastly, Links table stores incoming and outgoing links associated with a particular card. From the database point of view, it provides us with an overview of exchanged data in the module. However, the complete database is presented in Annex G. In order to depict the structure of knowledge card system, the UML class diagram of the system is illustrated in figure V.8.

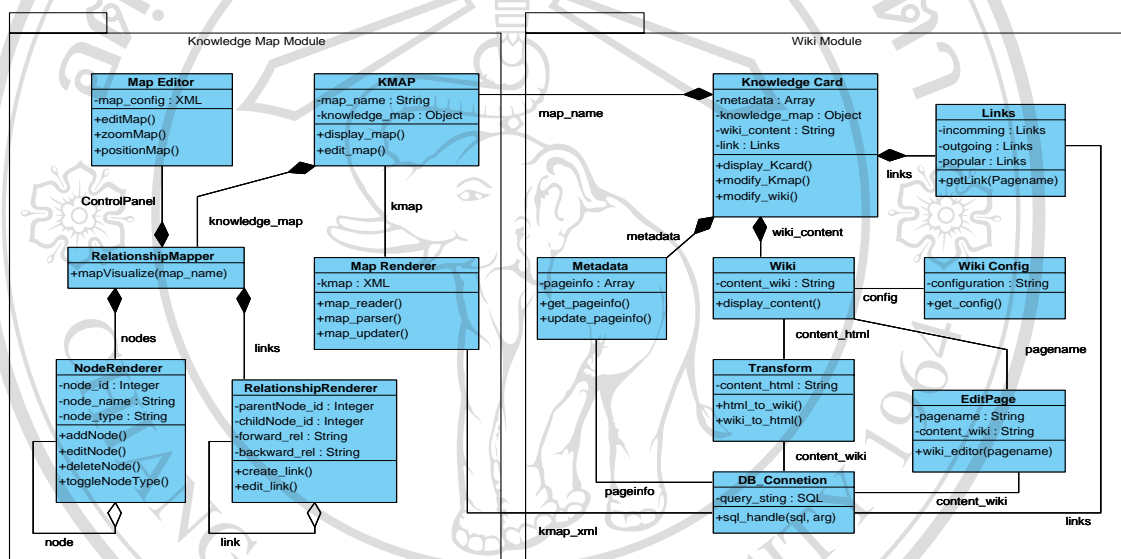


Figure V.8: UML class diagram of knowledge card system

The diagram is composed of two modules, knowledge map module and wiki module. Wiki module is the center of this system. The lowest level of the system is database connection (DB_Connection) class. This class is handling all connections between the knowledge system and SQL database. The Transform class is in charge of transforming wiki content from the DB_Connection into HTML content. On the contrary, it is also used for transforming HTML input from user into wiki content in order to store in the database. The EditPage class is a module that allows experts to edit wiki content in the knowledge card. The Wiki and WikiConfig modules concern the user interface of the wiki content in the knowledge card. The objective of these classes is displaying formatted wiki content on users' HTML browser. Finally, the

information from Metadata, Wiki, Links and Kmap class are integrated in the knowledge card class so as to display the required knowledge to knowledge workers.

Knowledge map module is an important element in the knowledge card system. It provides the body of knowledge about the task to the knowledge worker at first glance, without him having to read the content inside the wiki module. Moreover, it was employed by the inference engine for enabling ontology search in the knowledge system. The structure of knowledge map module is composed of two main classes, MapRenderer and RelationshipMapper. MapRenderer handles the connection with database in wiki module. The knowledge map in XML format is manipulated in this class. RelationshipMapper aims at visualizing knowledge map and its control panel on the knowledge card. A relationship manager comprises three elements, control panel, nodes, and links. Control panel is rendered by MapEditor class which allows the user to edit, zoom in/out, position map, etc. Nodes are rendered by NodeRenderer class which aims at representing task and domain knowledge in graphical format. RelationshipRenderer focuses on rendering inference knowledge on the knowledge map. This class draws an arrow between two nodes in order to indicate the forward relationship from parent node to child node. Although backward relationship is invisible to the reader, it is used by the inference engine for browsing the knowledge map. In order to emphasize the structure of the knowledge map, user interfaces of the knowledge map will be presented in this section. The captured screen of the main screen of knowledge map is illustrated in figure V.9.

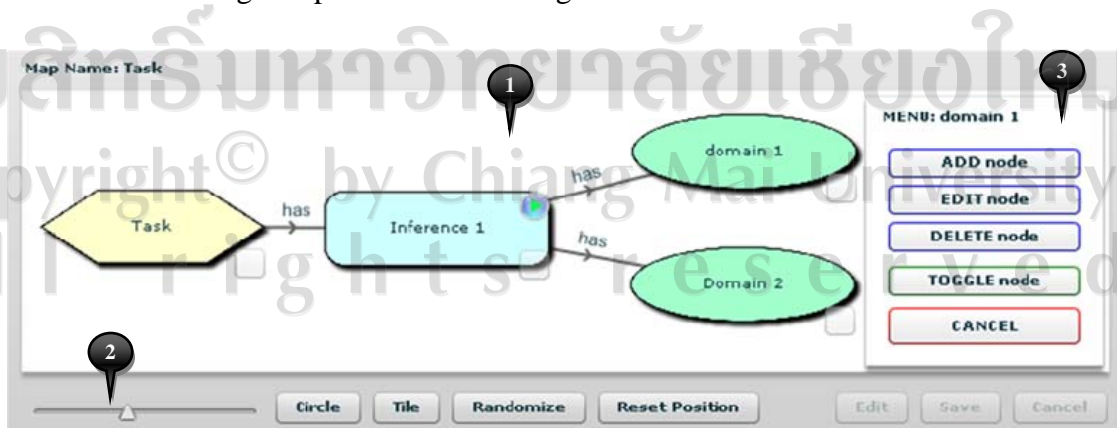


Figure V.9: User interface of knowledge map module

Actually, the knowledge map will be displayed within the knowledge card. However, the figure above focuses on the knowledge map component only. The knowledge map component can be divided into three sections: map area (1), control panel (2), and menu (3). The map area is a section for displaying knowledge map in the format of a semantic map. Users can drag each node or move the view pane easily. The control panel section allows readers to rearrange the map style, zoom-in/out, or enable the edit mode of the map. When the edit mode is enabled, the menu would popup on the map area. This menu enables the reader to add, edit, or delete nodes from the knowledge map. In addition, it is able to alternate node type between task and domain node. The updated map will be stored in the database when the Save button in the control panel is clicked.

In summary, this section presented major components of the knowledge card, the basic element for representing and sharing knowledge among the cluster members. Thus, the notion of knowledge card was considered as a medium for sharing the task intensive knowledge in this study. Moreover, the knowledge creation and representation techniques were proposed in order to create the knowledge model for the knowledge card. Finally, the user interface was illustrated for demonstrating knowledge card manipulation. In the next section, we will demonstrate the usage of the knowledge card in the system.

V.3.2. Knowledge Search

The knowledge search is a module that focuses on reusing the knowledge which is stored in the knowledge system. The objective of this module is to provide requested knowledge from the knowledge system to the applicant when needed. Compared with knowledge management activity, this module supports the knowledge reuse/retrieval facility. In this section, we will describe the user interface of the search module, which is composed of two views. Then, an algorithm of knowledge module will be depicted. Next, two knowledge acquisition techniques, which are the fundamental of this module, will be described along with knowledge acquisition scenario. These will give a better understanding about the usage of the knowledge.

In order to describe the overview of the knowledge search module, the user interface of search module is illustrated in figure V.10. The user interface of this module is separated into two parts, *list view* and *map view*. Both views provide an alternative for knowledge users to display the retrieved knowledge in two formats. List view is displaying the search result in a format of listed items, while map view displays search results as the knowledge map concept. For example, the list view in figure x shows the result from searching with the keyword “Purchase”. The result sets were listed with links to the knowledge card. The map view shows the same result in the different way from list. The map view in the figure implies that one knowledge card which contained the keyword is found which is represented by a rectangular symbol. And, it also found a group of concepts which relate to the keyword, represented by an oval symbol. In order to view the detail, the user is able to click on the transparent square button. Then, the second level of the knowledge map will be displayed.

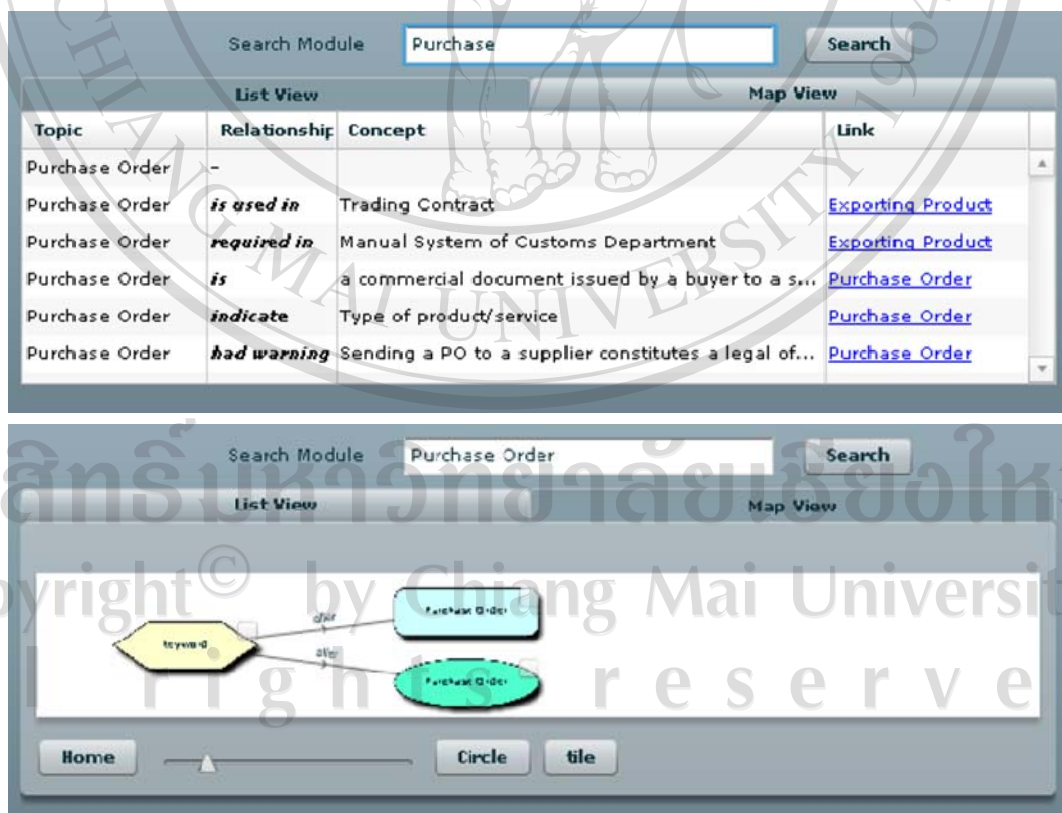


Figure V.10: Search results in the List and Map view

The *list view* is effective when the result set contains large amounts of search results. This view aims at providing all search results and supportive information within a single page. The result list is composed of four columns: topic, relationship, content, and page name. The first column (topic) shows the list of matched concepts. The second column shows the relationship between the topic and the knowledge content - which is displayed in the third column. The last column shows the page name (or card name). The user is able to click on the page name which will link to the knowledge card. In this view, the knowledge worker is allowed to customize their search results by re-sorting or eliminating unwanted results from the list.

The *map view* displays retrieved knowledge in the graphic mode. The map view is divided into two levels. The first level of map view shows groups of concepts that have the same topic, and also the card name that exactly matches with the search keyword. From this level, the user is able to access directly to the matched knowledge card (if found), or open the second level that will display domain knowledge about the selected topic. In this level, the user can acquire knowledge about a particular topic in the form of semantic map. Moreover, the user is allowed to jump to the knowledge card that contains particular domain knowledge by clicking on the map link. This view is suitable when the search results are not too large.

In order to describe the searching process, the algorithm of search engine is illustrated in the form of a flow chart, as shown in figure V.11. The flow chart comprises two parts, querying and displaying. The beginning of the flow chart concerns the querying method. Then, the final part concerns displaying the retrieved knowledge in list or map view. From the flow chart, we can see that the knowledge acquisition process commences when the user sends a search keyword to the system.

There are two type of querying process, querying via database connection module (content search), or querying via the inference engine. As soon as the result set was received from the database, the displaying process is initiated. If the search keyword could not be found in the database, the returned result set is null. Then, the system sends a message to the user that the requested page was not found, and then terminates the search process. But if the result set is not null, it will be sent to the displaying module which has two viewing options, list view and map view. The user is able to choose their search display option according to their preference.

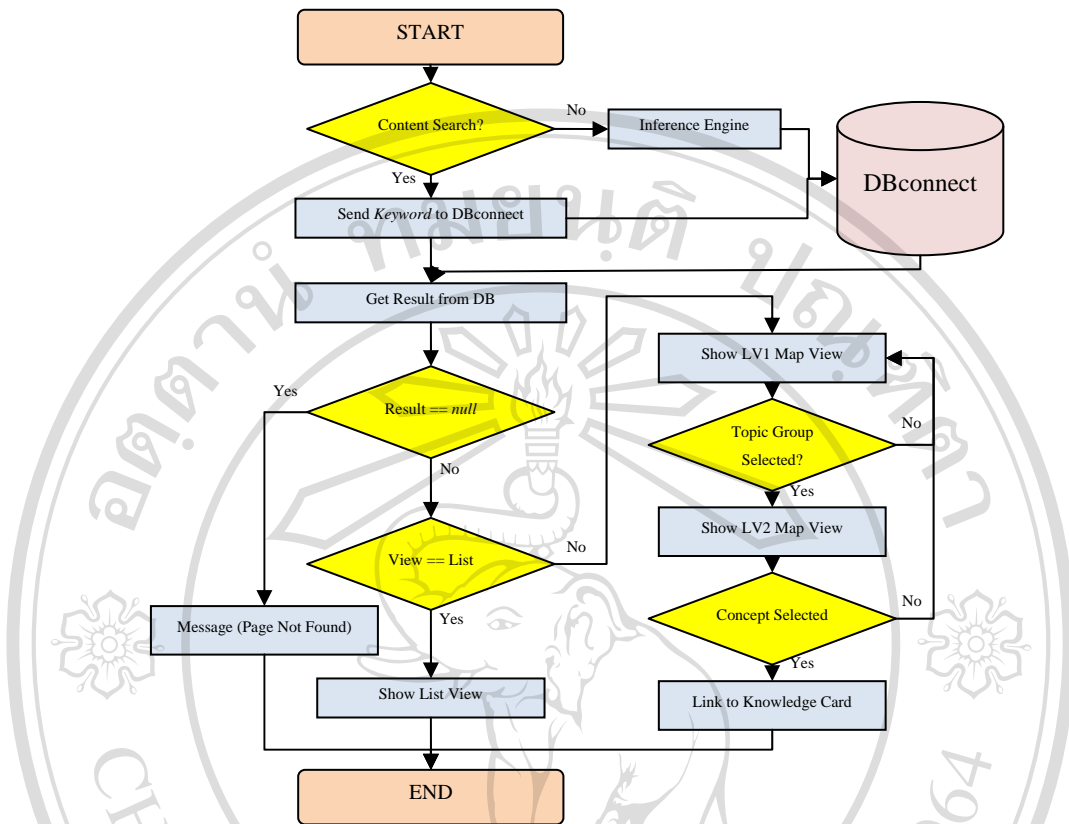


Figure V.11: Flowchart of knowledge search module

As we mentioned earlier, the user is able to choose one from two types of search method, content search and semantic search. *Content search* is an ordinary text search. It allows the user to search for the knowledge card that contains the search keyword anywhere in the card. The matched cards which include the keyword in their card will be accumulated in the result.

Another type of knowledge search is searching over the concepts on the knowledge map, called a *semantic search*. This type of search is adapted from the concept of the ontology search which makes it possible for the inference engine to search through the semantic relationship between the concepts. Semantic search gives fewer number result sets, but more precise search results than content search. The key factor that enhances capability of the search engine is the semantic relationship within the knowledge map. It facilitates the acquisition by integrating search results from various knowledge cards, and generates a new semantic concept map, called the result map. In order to generate the result map, the two techniques of forward and backward reasoning are adopted in the semantic search module. Forward reasoning allows the

knowledge user to search over the domain knowledge with a keyword and inference in the knowledge card.

The result from the search engine is a knowledge map that contains the domain knowledge elements which relate with the keyword. In algorithmic view, the inference engine browses all knowledge cards in the system and searches for the domain knowledge which matches with the keyword. Then, the related nodes and their relationship from different cards are integrated in order to create a new knowledge card which explains the keyword. Figure V.12 demonstrates an example of knowledge acquisition by using the forward reasoning method. On the other hand, backward reasoning is an inference method which intends to find the consequent from the list of goals. For example, the knowledge user searches with a set of keywords. Then, the inference engine looks for possible nodes which relate to the search keywords. Finally, the node that exactly matches with the keywords will be placed on the top of the list while the partial matches will be listed later on.

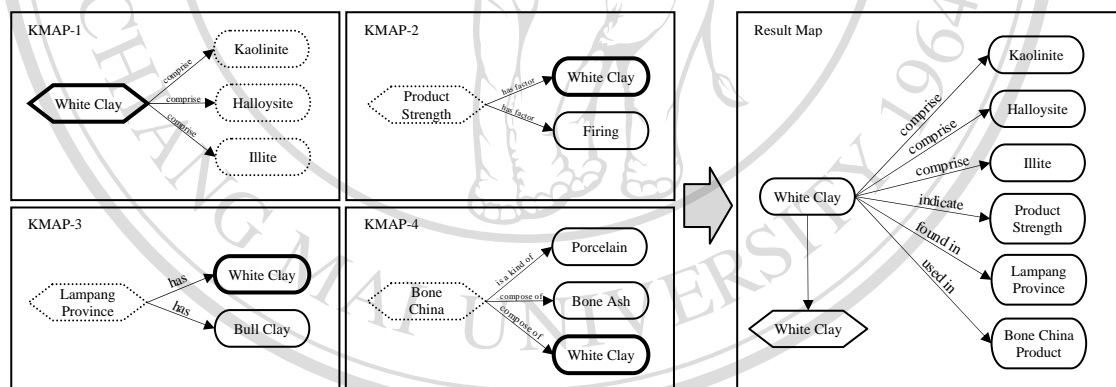


Figure V.12: Knowledge card searching and the result map

From the given example, a knowledge user acquires the knowledge about “white clay” from the knowledge system. Consequently, the inference engine found four knowledge cards (i.e. KMAP- 1 to 4) that contain the search keyword. The result set implied that the search module found one knowledge map which exactly matched with the search keyword (KMAP-1), and three knowledge cards which contain the search keyword in their domain concept (KMAP-2 to 4). The matched concepts were represented with a bold border. Then, it browsed over the corresponding nodes via forward and backward inference. The corresponding concepts were represented with

dash borders. An example of forward inference is the knowledge model in KMAP-1. This map explains the composition of the white clay, which is composed of Kaolinite, Halloysite, and Illite. Conversely, the backward inference was used in KMAP-2 to 4. The inference engine employed the backward relationship to indicate the link between matched node and corresponding node. For example, the result obtained from employing backward inference on KMAP-2 could be “White clay is a factor of product strength”.

The result map is composed of a link to the knowledge card that exactly matches with the keyword and the corresponding concepts. The links allow the user to access a specific knowledge card directly. The related concepts were attained for generating the result map. From the example above, the result map was represented in semantic concept map format, which could be read by the human as “*White clay comprises Kaolinite, Halloysite, and Illite. It indicates the ceramic product strength, can be found in Lampang province, and was used in bone china products*”. The result map was also translated into XML format which makes it possible for the knowledge system to exchange information with other application. To demonstrate the application of the knowledge card, the application section will describe how to make use of the knowledge in the system, and how we enhance the knowledge acquisition module with the mobile technology.

V.3.3. Knowledge Exchange Scenario

In the present situation of the ceramic cluster in our case study, there are neither tools nor systems that support knowledge management in the organization. However, the development of the industry cluster was maintained by the capability of the facilitators, support from the government, and the opportunities inside the industry cluster. Although these factors kept the members of the group together, they still faced the problems of knowledge sharing among members. For example, the ceramic trade fair exhibition is one of the most important business activities of this cluster due to it provides a great opportunity to the companies to access new markets and new customers. In order to achieve this, various type of knowledge are required for making decisions about this event such as product selection for the exhibition,

logistics planning, exporting, acquiring support from the government, booth preparation, etc. Thus, the ceramic cluster had organized the meeting to share information and make decisions about this activity. Unfortunately, the required knowledge and information never satisfied the members due to many reasons. For instance, the expert was absent from the meeting, the knowledge and experience of the previous trade fair were not in explicit form, the documents had been mislaid, etc.

Another critical problem of knowledge sharing in the cluster is the competitor-like relationship. This relationship sometimes makes experts feel uneasy to share their knowledge with their competitors. These are the obstacles to the potential achievements of the industry cluster. Hence, the proposed KMS is considered as a solution for solving these problems. This part of the research will clarify how the knowledge system will ameliorate the problematic situations in the ceramic cluster. The UML sequence diagram in figure V.13 demonstrates a complete scenario of the cluster members from requesting, sharing, and reusing knowledge. The ceramic trade fair event was selected as a case study of knowledge sharing. This scenario was divided into three periods of time, before, during and after the trade fair.

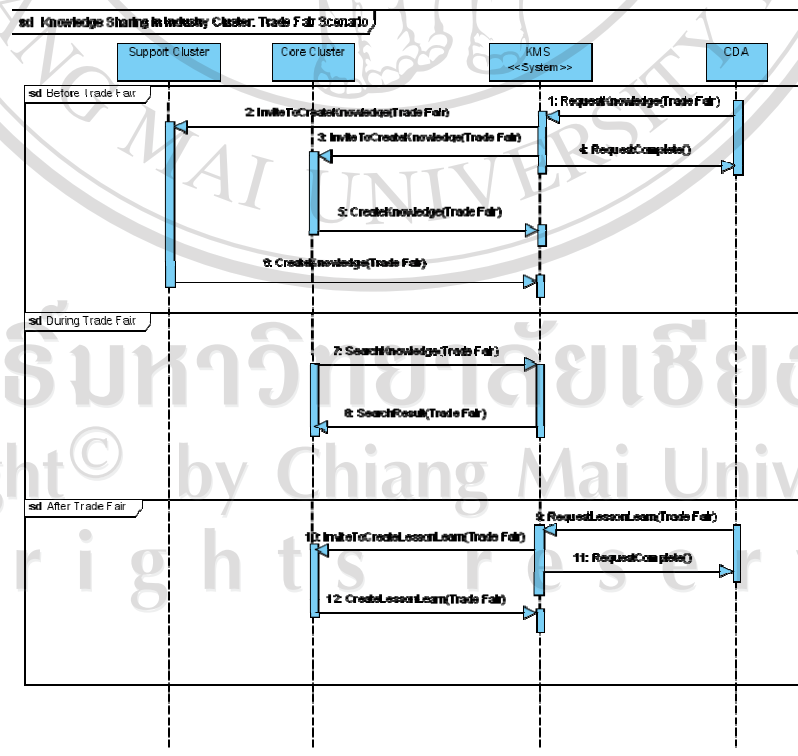


Figure V.13: UML sequence diagram of knowledge sharing scenario

In the beginning of the scenario, a government agency offered an opportunity for SMEs in the ceramic cluster to join the international trade fair in the form of some subsidies from the government. Collaboration services were used to present this opportunity to the cluster via the calendar module. After that, CDA sent a message to all members, inviting experts to create a new knowledge base in this CoP. The process was done by Push News system on the KMS. Cluster members received an invitation, with URL link, via e-mail in order to contribute to the new knowledge card. Experts who had experience about the international ceramic trade fair got access to the knowledge card module and shared their knowledge in the format of knowledge card. The advantage of this system was in allowing experts to represent their knowledge over other knowledge. This helps experts to put less effort into sharing their knowledge, and focus only on their domain of expertise. This knowledge sharing process is illustrated in figure V.14.

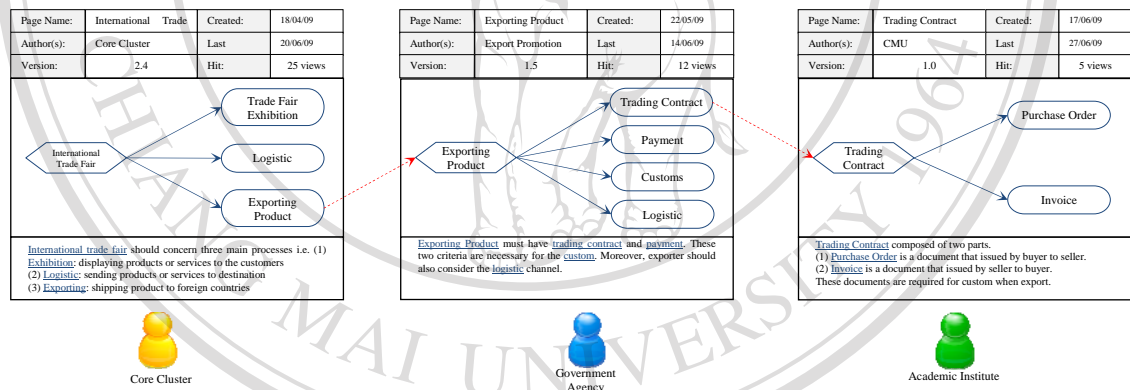


Figure V.14: Knowledge sharing via Knowledge card module

The figure above demonstrates that experts from different domains can contribute their knowledge on a common field via the knowledge card system. In our scenario, these experts were invited to share their knowledge on the knowledge system. An expert from the core cluster logged into the system and provided the body of knowledge about the “*international trade fair*”. Then, an expert from the Department of Export Promotion (a Thai government agency) accessed the system. He appended his knowledge about “*exporting product*” to the previous knowledge. Finally, an academic institute added a new knowledge card about “*trading contracts*” to the system. The knowledge system automatically linked these knowledge cards

together with the inference engine. This process portrayed how the knowledge base was constructed in the KMS.

During the trade fair, a knowledge worker at the exhibition required knowledge about “trading contracts”. He searched with the specified keyword in order to request knowledge through the search engine. The inference engine searches for a knowledge card which matched with “trading contracts”. Then, the result set was returned to the inquirer in the format of knowledge card. The procedure of searching was previously described in the knowledge acquisition section. The result set obtained from searching in the knowledge card above is displayed in figure V.15.

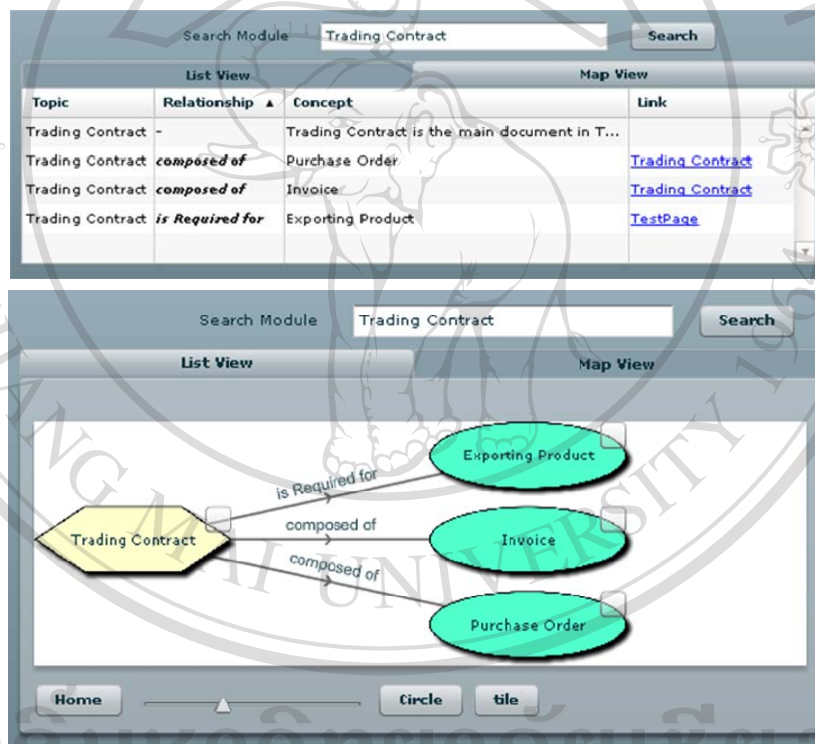


Figure V.15: User interface of search result module

The search result obtained can be displayed in two views, list view and map view. From the figure above, the top side shows the search result in list form. List view displays complete information about the trading contract. Conversely, map view displays search results in the knowledge map model. The outcome from this page can be read as follow; “A *Trading Contract* is required for exporting product. It is composed of Invoice and Purchase order”.

knowledge model of the model suite which has been depicted in Chapter 3. The particular knowledge about a task is transcribed and represented into the structured knowledge model and add to the knowledge base directly via the knowledge acquisition module. This module is accessible only by the administrator, CDA or knowledge engineer, which is the main module for maintaining the knowledge.

The second approach is transferring the knowledge directly from the cluster members to the knowledge base via the KMS. This method allows the experts in the cluster to share their knowledge to the cluster liberally. However, the ontology module is implemented to the system in order to maintain the consistency of the system. The general objective of the ontology module is that it checks the relationships between the related concepts. It ensures that the backward relationship is appropriate with the defined forward relationship. The objective of the task ontology module is to verify the new knowledge concept whether it is existing in the knowledge base. If the existing concept is Task knowledge (is a name of knowledge card), it will change the symbol of new concept to “Inference” symbol. But if the existing concept is Domain knowledge, it will link new concept with the group. Thus, the inference engine is able to browse over the concept during the searching process. The objective of the domain ontology module is to provide the common vocabulary about a specific domain. For example, CDA and Cluster Development Agent are defined as the same topic in the ceramic cluster domain. Thus, the inference engine is able to show the combined result map from the result sets of both concepts.

V.4. Supporting Services

The main objective of these supporting services aims at sustaining knowledge management activities of the industry cluster. Although these services are not focusing on the knowledge itself, they permit the cluster members to exchange the knowledge within the cluster with different approaches. From the study of [Malhotra 04], many KMS have failed to integrate in the organization due to neglecting this issue. Thus, in this study, we propose two types of services i.e. presentation and collaboration service in order to support knowledge management activities in the

ceramic cluster. These services will be added to the KMS in order to enhance the functionality of the system.

V.4.1. Presentation services

The presentation service was designed with the intention of visualizing and personalizing all services provided on the KMS to the users in an appropriate way. The main objective is to harmoniously integrate the KMS into cluster activities. Moreover, it was set to facilitate cluster members to share and retrieve knowledge with less effort. This service aims at integrating the knowledge system's services with the activities of cluster members. The failure of many knowledge systems are caused by neglecting the integration of a developed system with the actual activity of the organization. Experts may need to double their efforts in order to share their knowledge with the knowledge workers in the workplace, and also the knowledge system. This situation will decrease willingness to share organizational knowledge. The benefit of the presentation service is in allowing users to personalize their view for presenting the information or services from the knowledge system. In order to achieve these goals, we have created two applications for supporting presentation services, *KMS portal* and *Widget*. The details of both applications will be described in the following Figure V.17 which illustrates how the KMS portal and knowledge widget were integrated in the KMS server.

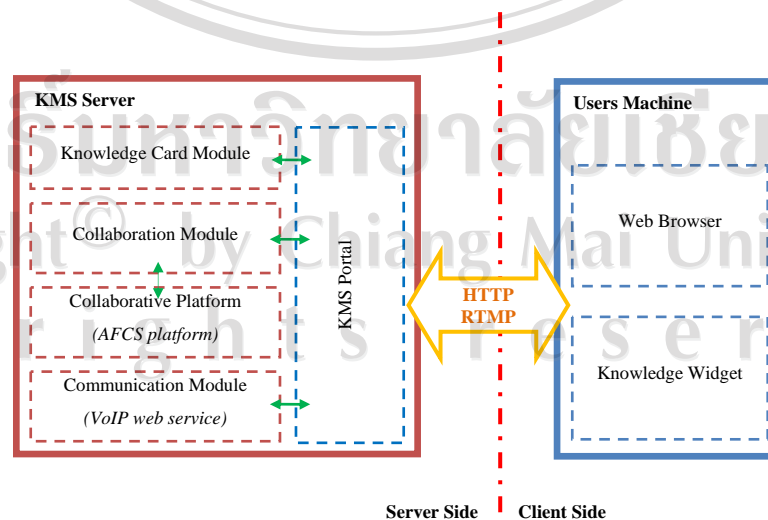


Figure V.17: Presentation services and KMS server integration

The architecture revealed the integration of presentation services in the knowledge system. The modules for presentation service were annotated using blue, while the KMS system was annotated in red. The presented architecture implied that there are two manners of application that support the service, server side and client side. The KMS portal is a server-side application which allows users to access all provided services on the KMS via web browser on the client side section (e.g. Internet Explorer, Mozilla Firefox, etc.). The KMS portal is a server-side application which aims at presenting services from diverse modules (e.g. knowledge card, collaborative tools, communication, etc.) in a unified way. Portal also handles HTTP connection from web browser on the users' machines. Another application is the knowledge widget, which was developed for this knowledge system. Adopting the notion of widget in the system will facilitate cluster members to get direct access to all available services on the system. Besides, it makes it possible for knowledge system to push information to all users, and conversely allows users to pull information from the knowledge system. Although these modules are mainly concerned with the user interface, it is a key achievement of the knowledge system. Cluster members learn the benefits from the knowledge system via KMS portal and knowledge widget. The following parts will give explanations on both modules.

V.4.1.1. Knowledge Management System Portal

The KMS portal provides a single point access to a variety of contents and services on the knowledge system, and ideally offers a single sign-on point. The portal often organizes information into channels, customizable page containers where specific information or an application appears. Another major benefit of the KMS portal is that it offers an access point for users in the same community to collaborate. Since the concept of second generation of web development and design (called Web 2.0) was popularized in recent years, every web portal has tried to facilitate communication, secure information sharing, interoperability, and collaboration on the World Wide Web. The difference between Web 1.0 and 2.0 is that Web 1.0 is about linking documents, but Web 2.0 is about linking people together. Thus, knowledge management practice has taken an advantage from Web 2.0 by supporting experts and knowledge workers in the organization to collaborate and exchange their knowledge

on the particular practices. Although there is no defined architecture, the KMS portal is designed with respect to the organization requirements and provided services of the system. Thus, the site map of KMS portal in our study is proposed in figure V.18 for presenting the overview of portal's architecture.

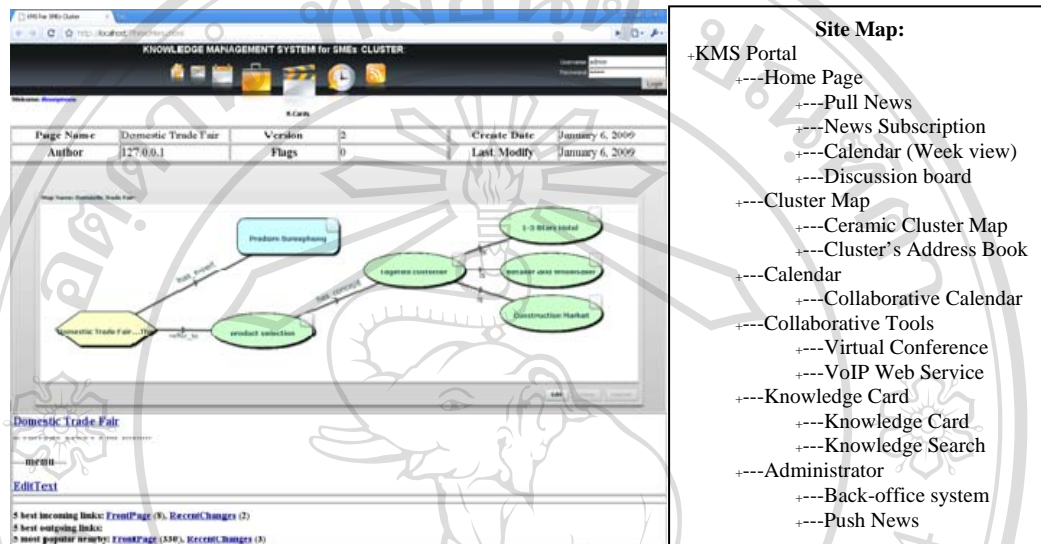


Figure V.18: KMS portal page and site map

The architecture of proposed KMS portal is composed of six sections: portal, cluster map, calendar, virtual conference, knowledge card, and administrator. The portal's screen is illustrated on the left hand side of the figure. It comprises two parts i.e. menu and content part. The menu part is located on the top of the screen which allows users to access to the provided services on the knowledge system. Content part is the main panel of the screen. Each content module was placed according to the significance of the information and preference of the users. The example shows the knowledge card page on the KMS portal.

The site map implies that users are able to access other services of the knowledge system via the top menu. The cluster map menu gives access to the cluster map and address book service. The calendar menu shows full function of the collaborative calendar service. Then, the collaborative tools menu allows users to employ all virtual conference services and VoIP web services via the web browser. Knowledge card page contains two main services of the knowledge system,

knowledge card and knowledge search. Lastly, the administrator menu links to a secure page that allows the CDA to manage the KMS portal and underlying services.

In order to access the services over the KMS portal, users have to identify themselves via login module. An anonymous user is permitted to access only the home page and cluster map. Referring to Info-Structure in chapter 4 (figure IV.7), cluster members agreed to share this information outside the cluster. Identified users are allowed to access all services on the knowledge system. However, some modules were personalized in regard to their roles in the industry cluster. For example, a user from the core cluster will be connected with “Ceramic Production CoP” automatically when they logon to the system, but a user from a government agency will be connected with Export Promotion CoP instead. This concept also refers to the Info-Structure that users are more willing to share specific knowledge if they are in the same community of practice. CDA has the privilege of accessing some exclusive services such as Push News, back-office, user management, etc. The advantage of the KMS portal is providing an access point for cluster members to use all provided services on the knowledge system without installing any special software on the client side. However, the KMS portal is limited by HTTP service and web browser. Thus, the next section will propose the knowledge widget, which is a client side application. It makes it easy for cluster members to access all services on the knowledge management server from their own desktop.

V.4.1.2. Knowledge Widget

The notion of widget was well-known in recent years, due to the emergence of redundant information on the internet. In the past, knowledge management practice faced a problem about lacking of information to achieve the task. Nowadays, information is available anywhere on the internet. A newly emerging challenge is how to retrieve a minimum of information, yet enough to accomplish the task. Therefore, the concept of widget is adopted for retrieving information from different locations and presenting it to the user in an appropriate format via GUI. Widget is mostly used for displaying information which is often used. In this study, we create a widget for retrieving information from the different servers (knowledge server, collaboration server, and communication server), called the knowledge widget. Moreover, it is also

designed for facilitating cluster members to collaborate with others from a single point. In order to get a big picture of the knowledge widget, the structure of the widget is illustrated in figure V.19.

From the structure presented, the concept of widget was clarified with UML class diagram, as shown in figure V.20. The knowledge widget is composed of eleven classes. These classes included three service classes: Knowledge, Collaboration and Communication service, which was designed for handling the connection and authentication to the service provider. The main class that takes control of these service classes is Cluster Widget.

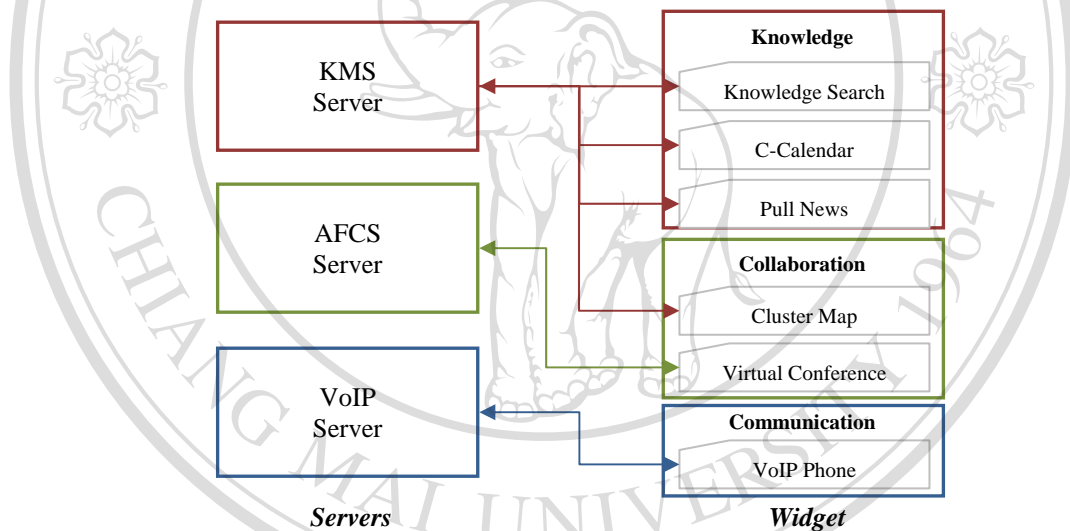


Figure V.19: Architecture of the knowledge widget

The Cluster Widget class is in charge of user authentication. As soon as the authentication is successful, the `widget_config`, which stored parameters for personalizing the widget will be retrieved from the Config class. Then, three associated service classes (knowledge, collaboration, and communication) logon to the servers and acquire permission to get access to the provided services. Three knowledge services (news, event, and knowledge search) were provided by the knowledge server directly. Collaboration services displayed cluster map from the knowledge server and connected with AFCS server.

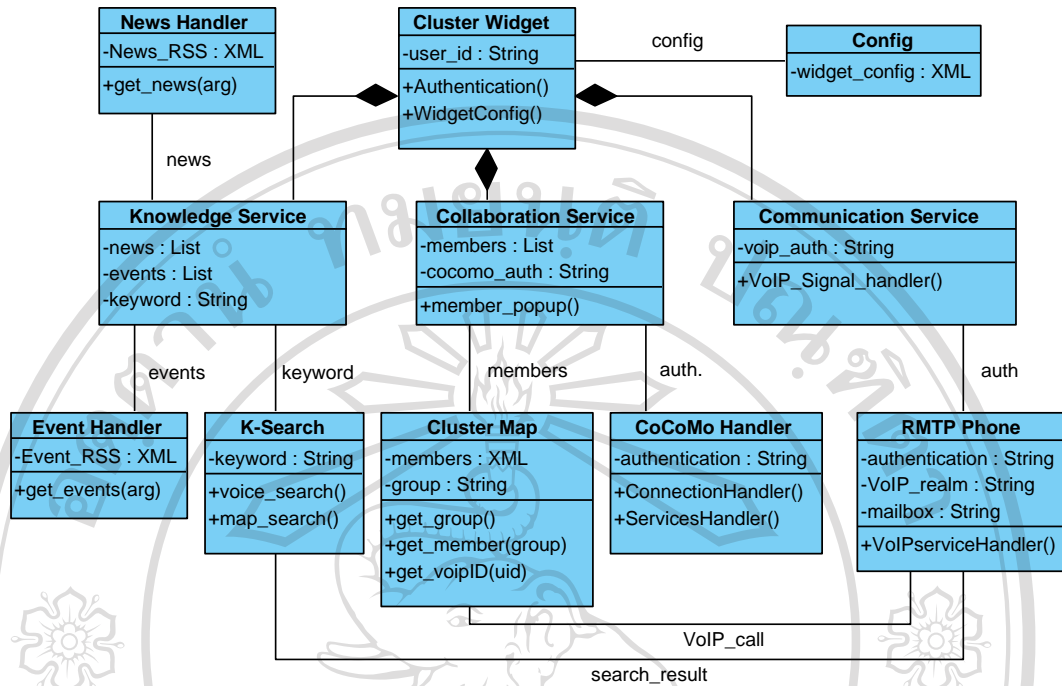


Figure V.20: Knowledge Widget’s UML class diagram

Finally, communication service class gets access to the VoIP phone web service on the knowledge server in order to allow users to connect with the communication server. With the intention of clarifying the functions of the widget, the captured screens of each service are illustrated in figure V.21.

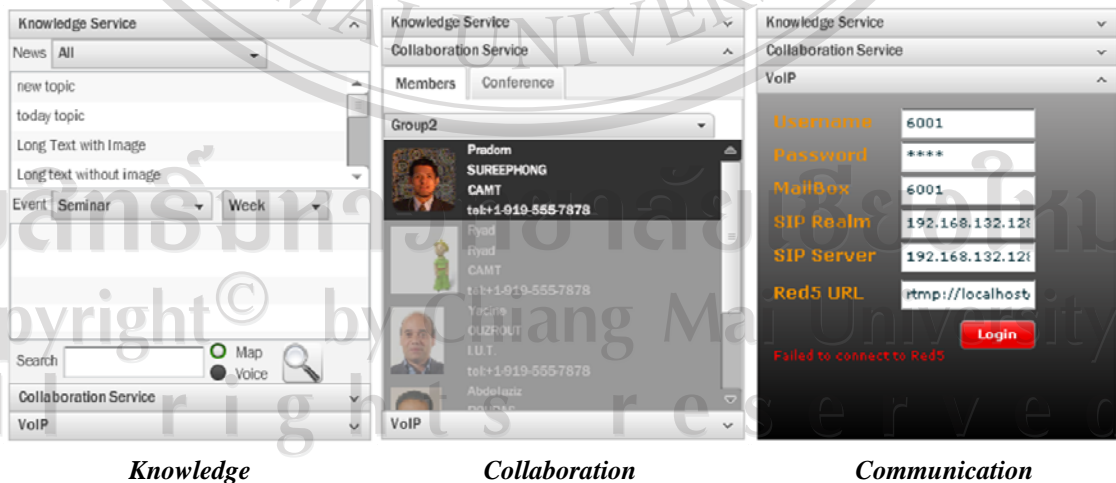


Figure V.21: User interface of the knowledge widget

The user interface of knowledge widget showed three main services of the knowledge system. The first screen that will be displayed to the user after logged in is *knowledge panel*. This panel handles three types of service from the KMS server i.e. pull news, c-calendar, and knowledge search. Pull news module allows users to retrieve update information from the cluster. The c-calendar is represented with the event module which allows users to view specific type of event (e.g. seminar, meeting, training, etc.) in the specific period (e.g. today, this week, etc.). Finally, the knowledge search module allows user to reuse the knowledge from the knowledge system. The knowledge widget provides two ways of search i.e. map and voice. If user selected map search approach, the search result will be transmitted to the user in the semantic knowledge map format via the web browser (which was explained in the previous section). But if user selected voice approach, the knowledge map will be synthesized into voice format. Then, the search result in voice form will be transmitted to the user via *communication panel* of this widget. The details of these processes will be described at the end of this chapter.

The *collaboration panel* allows user to connect to two services i.e. cluster map and virtual conference. The cluster map module provides the personal information of each actor in the ceramic cluster. Moreover, it also allows users to communicate by peer-to-peer approach. On the other hand, the conference tab facilitates the cluster member to collaborate in group. The details of these services will be described in the collaborative services section. These captured screens proved that user is able to access all provided services of the knowledge system via the knowledge widget. However, the next section will describe the usage of these collaboration services and their benefits to the cluster.

V.4.2. Collaboration services

This level of the service in the system architecture aims at enhancing quality of collaboration and communication of the cluster members. The analysis from collaboration model in chapter 4 found that the activities of the cluster member which have high degree of collaboration and high impact to the industry cluster could be categorized into three major activities i.e. (1) sharing opportunity (such as seminar,

trade fair exhibition, workshop, etc.), (2) joint working in specific topic (such as joint exhibit in the trade fair, joint research, aggregate demand, etc.), and (3) solving problem together. In order to support these activities, three levels of collaboration are concerned i.e. mass, group, and individual collaboration. Each level of collaboration aims at supporting the collaboration of the cluster members in different aspects. In our study, the collaboration service model can be represented as in the figure V.22. It shows a group of collaborative tools which are selected to support each level.

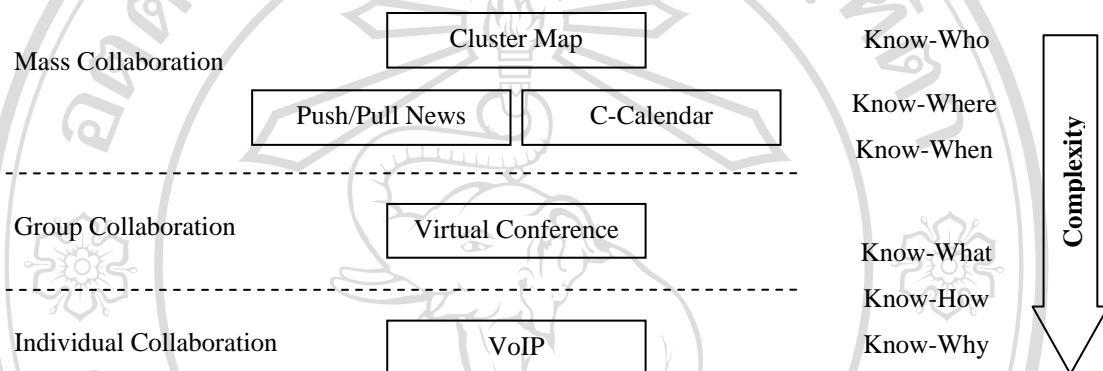


Figure V.22: Collaboration service model

The model implies the three modes of communication that are supported by the KMS server: mass, group, and individual (peer-to-peer). The knowledge system makes it possible for cluster member to broadcast their messages to all members via push/pull news system. This type of communication enhances the coverage of information sharing within the cluster. The virtual conference tools allows cluster member to interact within a restricted group of members. This type of communication improves frequency of collaboration due to it eliminates the barrier of collaboration in term of place and time. The final mode is the peer-to-peer communication which enables members to exchange information to each other individually. The objective of this service is to reduce cost of collaboration of the industry cluster by adopting VoIP technology in the system.

In order to provide a general view of the collaboration services of the KMS, the UML class diagram in the figure V.23 shows the elements and inter-connection between classes for these services. The diagram comprises three groups of classes. The first group is placed on the top-left of the diagram, called c-calendar. The second

group is the push/pull news which is placed on the top-right of the diagram. Finally, a group of cluster map classes is placed on the bottom.

In development aspect, the collaboration services can be divided into five modules i.e. cluster map, push/pull news, collaborative calendar, virtual conference and VoIP. The cluster map is considered as the center of these modules. It is accountable for managing cluster members' information, level of security, and authentication to other modules. Push/Pull news and collaborative calendar aim at improving sharing of information and opportunity in the cluster, while virtual conference and VoIP focus on ameliorating the communications among the members. The VoIP module will be described at the end of this chapter. We will first examine the cluster map which is the central module.

V.4.2.1. Mass Collaboration

The mass collaboration makes possible for cluster members to broadcast their message to everyone in the cluster or only to the focused group. The benefit of supporting this type of collaboration is that it improves the coverage of the information distribution in the ceramic cluster. Moreover, it is used for sharing some types of knowledge which are not in the complex form e.g. know-who, who-where, know-when, etc. Therefore, this part will focus on the collaborative tools which are designed to support these goals.

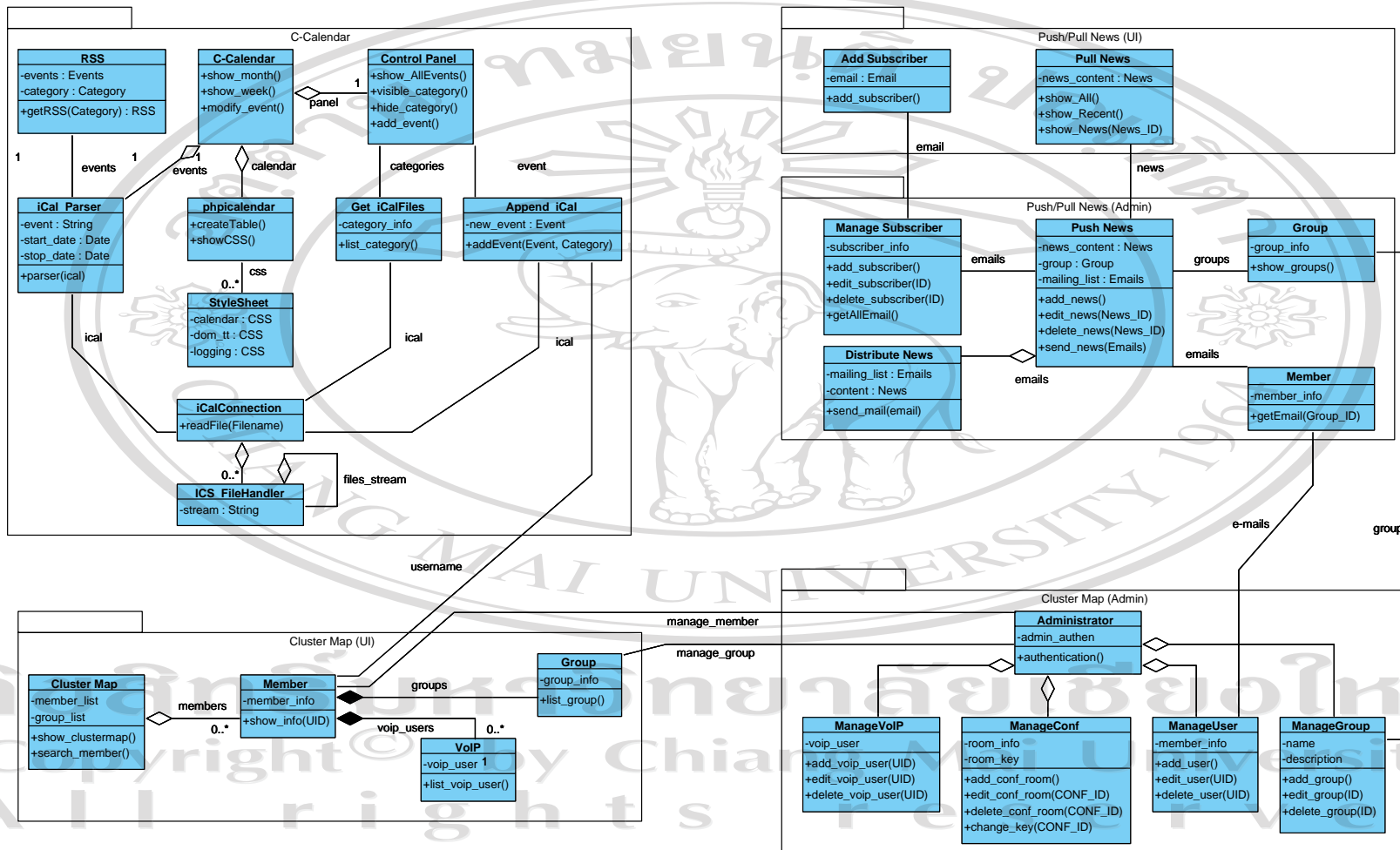


Figure V.23: Structure of the collaboration services in UML

(a) Cluster map module

The main objective of the cluster map is to provide the information of the industry cluster network in the visualized view. Moreover, this module contains all important information of the members for the system such as authentication information, user ID, e-mail, role, etc. Thus, this module became the center of collaboration level. The database of the cluster map module comprises four tables (Member, Group, VoIP_Auth, and Conf_Auth). The databases model can be found in Annex G. The fields in member table were customized from the ontology for vCards [vCard 08] to match with cluster context. The benefit of complying with the vCard ontology is that it provides the capability to share members' information with other systems. An example of a simple vCard is presented as follow.

```
<v:VCard>
  <v:fn>Pradorn Sureephong</v:fn>
  <v:url rdf:resource="http://pradorn.example.com"/>
  <v:email rdf:resource="mailto:pradorn.sureephong@univ-lyon2.fr"/>
  <v:photo rdf:resource="http://pradorn.example.com/images/pradorn.png"/>
</v:VCard>
```

This vCard is generated from the cluster map database and exported into XML format. The group table contains the group of member in the cluster (e.g. core cluster, academic institute, government agency). VoIP_Auth and Conf_Auth contain the required authentication information of the user for connecting with other servers. For a better comprehension, the class diagram in figure V.23 shows their interrelationship, attribute, and operation of the classes. The class diagram showed that the cluster map was composed of two modules: user interface module and administrator module. The administrator module focuses on managing information of each table in the databases, while the user interface module focuses on displaying information from databases to the users. It visualizes the members of the cluster by categorizing into different views such as group, roles, or CoP. The figure V.24 show cluster map which is categorized with the group view.

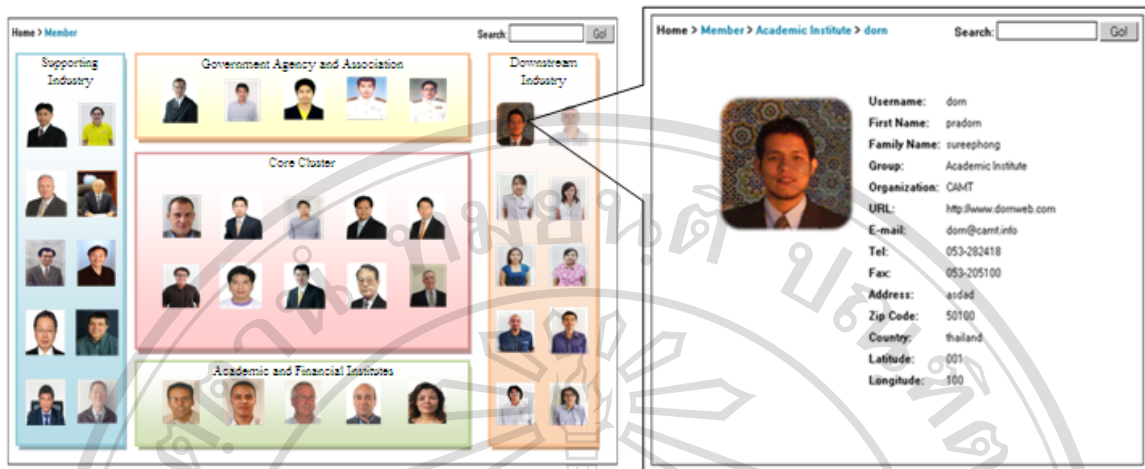


Figure V.24: Cluster map in group view with user's profile

The cluster map (left-hand side) shows an example of the groups of members involved with the ceramic cluster. It also allows users to export viewing result into vCard format for introducing to other applications such as Microsoft Outlook, Palm contact, etc. Moreover, users are able to view the complete information about each member by clicking on the member's image. The information of the user will show up as illustrated on the right hand side of the figure. It provides knowledge about the cluster network, called *know who*. The question about "*how do I know who knows?*" is a good example of accessing to the cluster map.

(b) Push/Pull News

One of the critical problems of the ceramic cluster is the thoroughness of information distribution. As we mentioned earlier that the cluster organization is flat, sometimes the delivered information may not go throughout the cluster, whilst sometimes CDA does not receive that information. The Push and Pull approach is adopted to resolve this kind of problems. Push News allows CDA to 'narrowcast' or rather to 'broadcast' personalized news directly to readers. It also empowers readers by letting them specify what content they want delivered, via which channel, as well as how often. Information and communication technology offers various channels to convey news to cluster members such as e-mail, voice mail, short messages, etc. On the other hand, The Pull technology is a style of network communication where the initial request for data originates from the client, and then is then responded to by the server. Pull News allow cluster members to access all news from the

server. News feed is an example of pull technology. It is published news on the server in RSS (Really Simple Syndication) format which end users can register with an aggregator program (e.g. feed reader or news reader).

The database schema of push/pull news is showed in Annex G. It shows two tables that are required in this service. The News table contains information about the news such as topic, content, image, etc.; which was designed according to the RSS 2.0 format [RSS 09]. The Subscriber table stores e-mails of the subscribers who are not a member of the ceramic cluster, but desired to receive information from the cluster. In order to visualize the usability of these modules, the GUI of these classes is illustrated in figure V.25.

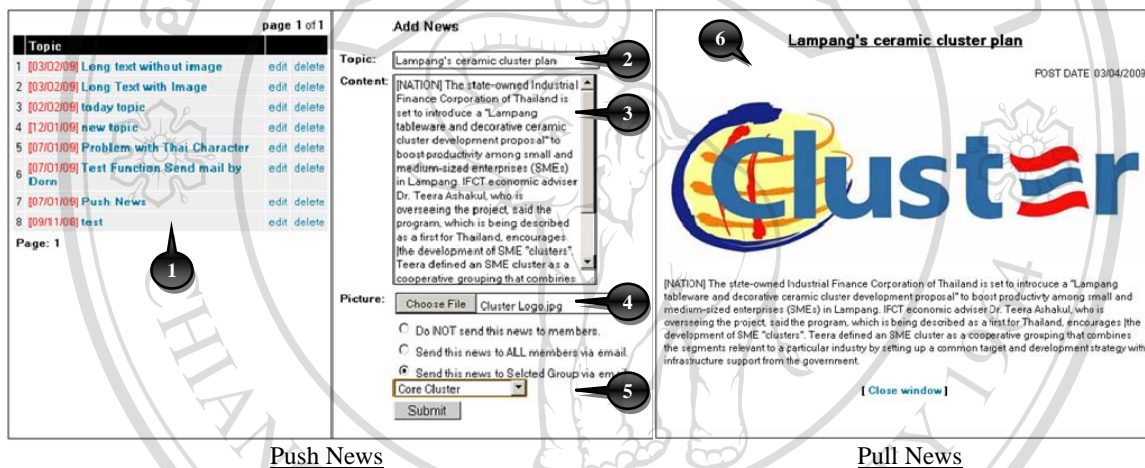


Figure V.25: User interfaces of Push and Pull News module

The Push News is displayed on the left-hand side, allows only CDA for narrowcasting news to a specific group of member. The Pull News can be accessed by cluster members and anonymous users in order to obtain latest information from the industry cluster. The left pane (1) displays a list of recent news in the system. It also allows the CDA to edit or delete out-of-date news. The right pane shows the form for adding new news. The CDA is able to add topic (2), content of news (3), and images (4) to be attached to the news. The radio boxes at the bottom of the pane provide the options for distributing particular news to the members. The Pull News interface (6) shows the screen that readers will see when they get access to the news system. The example in figure x demonstrates a scenario that the CDA would like to push information about “Lampang ceramic cluster plan” to the members in the core cluster. As soon as the CDA has clicked on the submit button, the news distribution class will send

the topic, content, image link, and news link to all ceramic producers in this industry cluster. At the same time, this news is stored in the news system and waits for Pulling by the reader. This system ensures information sharing, and every member will receive the exact same information. Next, we will present a tool that supports opportunity sharing for the ceramic cluster.

(c) Collaborative Calendar

The collaborative calendar service is designed for enabling cluster members to manage and coordinate their opportunities in terms of place and time. In a theoretical aspect, the exchanged opportunity in the cluster is a type of knowledge, also known as know-where and know-when. However, the opportunity varies according to the characteristics of the industry cluster. The examples of shared opportunity in the ceramic cluster are cluster's appointment, meeting of cluster committee, seminar from academic institutes, trade fair exhibition from government agencies, low interest rate loans, etc. These opportunities are the main criteria that attract the members with the ceramic cluster. Thus, the collaborative calendar (c-calendar) is proposed to support this information sharing. A calendar standard called iCalendar (RFC 2245) is adopted in order to allow the knowledge system to send the events and tasks to the other systems via email, or sharing files with an ".ics" extension. The basic elements of iCalendar (iCal) are: event, start date, and end date. Here is a simple example of an iCalendar object, "*Cluster Yearly Meeting event which occurs on July 14, 2009 (UTC)*".

```
BEGIN: VCALENDAR
VERSION: 2.0
PRODID: -//Apple Computer, Inc//iCal 1.5//EN
BEGIN: VEVENT
DTSTART: 20090714T000000Z
DTEND: 20090715T000000Z
SUMMARY: Cluster Yearly Meeting
END: VEVENT
END: VCALENDAR
```

The code above shows the basic syntax of iCal format. The texts in red color represent the value of the object, while the black color (e.g. VEVENT) represents the predefined syntax. Actually, there are many different types of components which can be used in iCal standard, depending on the purpose of usage. In general view, the collaborative calendar

service can be separated into two main parts: user interface part and iCal handler part, as shown in figure V.23.

We will first focus on the engine of the service. The ICS_FileHandler class manages reading and writing the ICS files in the server. Then, iCalConnexion read stream file from previous class and stores in the memory. The iCal_Parser aims at converting iCal format to event variables for file reading operation. Conversely, append_iCal convert events variables to iCal format for file writing operation. PhpCalendar put all events from iCal_Parser into created table which is defined with CSS (Cascading Style Sheets) format, while the category of event obtained from ICS filename via Get_iCalFile class. The user interface part comprises c-calendar, control panel, and RSS classes. The c-calendar class generates graphic interface and also control panel to the user. Finally, RSS class allow user to export event(s) into ICS or RSS feed format in order to exchange with other system or software application. For better comprehension, the graphical interface of collaborative calendar was presented in figure V.26.

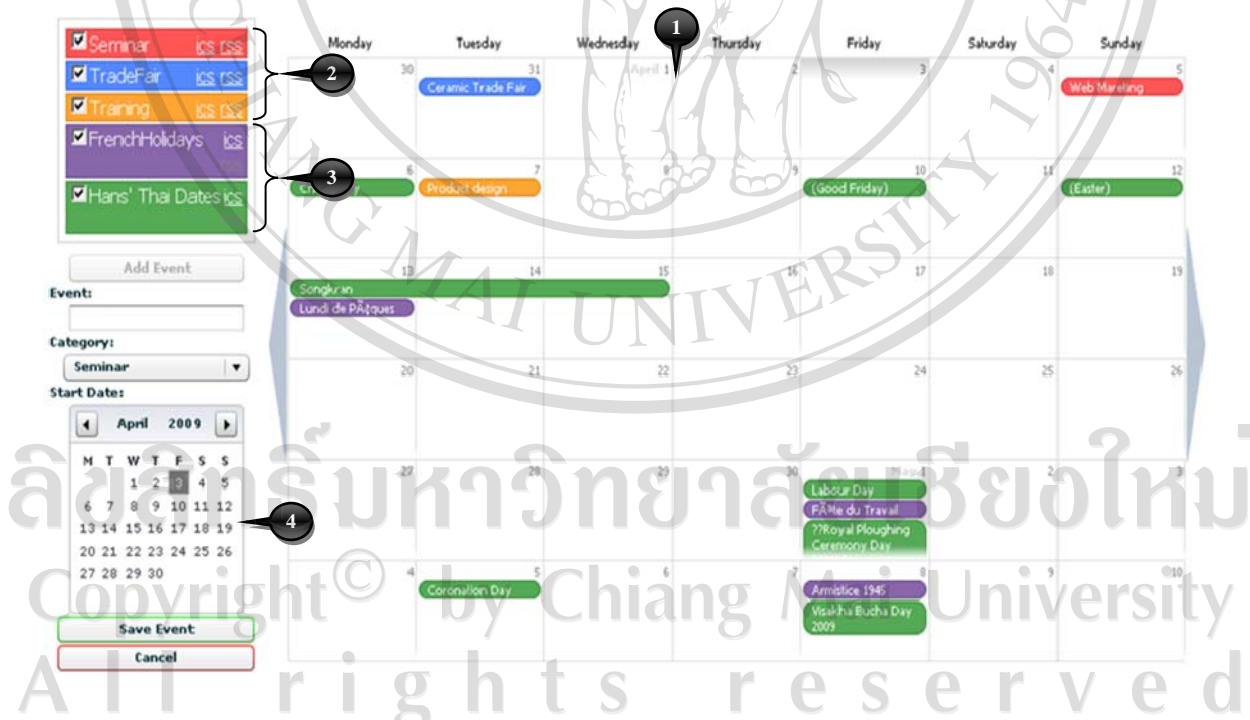


Figure V.26: User interface of the collaborative calendar

The c-calendar interface shows the events for the ceramic cluster by month. The main section (1) displays categorized event with different colors. This pane also allows a cluster member to create multi-day events or modify events by using a drag-and-drop approach. In

the control panel section, first three event categories (2): seminar, trade fair and training are ICS files on the server, while holidays events (3) are imported from Yahoo's iCal web service. Lastly, the bottom part of the control panel (4) allows the user to add new events to the calendar system. An example of opportunity sharing is that an academic institute gets access to c-calendar service in order to announce a seminar event on "web marketing". Two type of knowledge required in this module are *where* and *when* to get this opportunity. After that, the cluster member who is interested in this seminar could acquire further information about this opportunity from the description provided. Next, we will demonstrate how KMS can improve the quality of collaboration and facilitate knowledge sharing in the direct approach via *group collaboration* approach.

V.4.2.2. Group Collaboration

The group collaboration aims at supporting the cluster member to communicate or exchange their knowledge in a specific Community of Practice (CoP). This mode of communication allows experts to share more complex knowledge (e.g. know-how, know-what, know-why, etc.) to the knowledge users. The examples of this type of collaboration in the ceramic cluster are cluster meeting, group discussion, etc. Thus, the virtual conference module is designed to support this type of collaboration within the ceramic cluster.

(a) Virtual Conference

The virtual conference service aims at improving the quality of communication of the cluster member. This service allows members to communicate via text, voice or video through the KMS server. From the analysis, although a cluster meeting is organized monthly, less than 10% of the members of the ceramic cluster participated in this meeting. The main obstacles are place and time which are not convenient to the other members. Thus, this service tries to reduce these barriers and also increase frequency of communication at a lower cost. The main element of this service is the collaborative server (named AFCS) which was introduced in the beginning of this chapter. The platform is composed of five major components: session and authentication, shared manager, shared model components, collaboration user interface component, and pods. The organization of this platform is displayed in figure V.27.

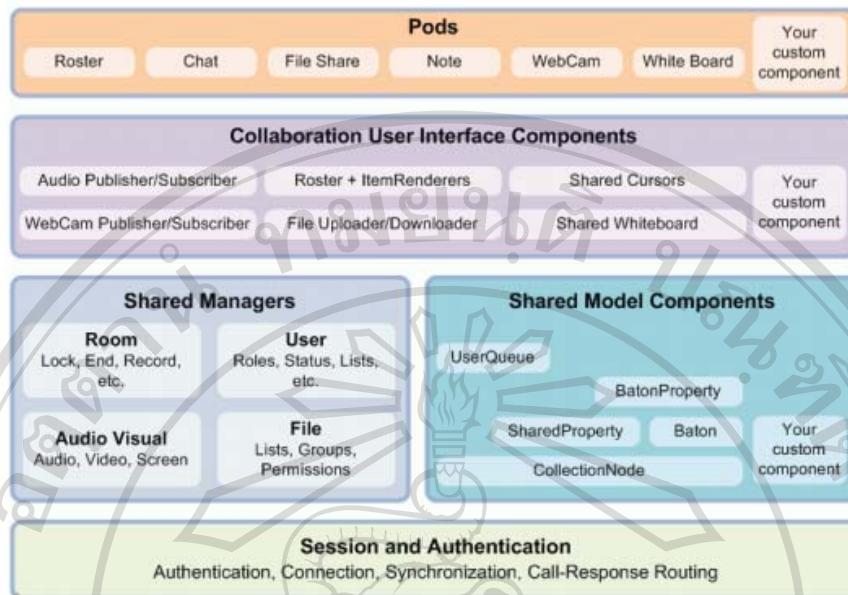


Figure V.27: AFCS architecture [AFCS 09]

- *Session and authentication* component handles connection and authentication of the conference members. Besides, it also controls the synchronization of the data between users' applications and the server.
- *Shared model* component involves the shared state of the application classes. This component manages user queue, shared property and permission of users in the conference room.
- *Shared managers* provide four pillars of every AFCS application.
 - RoomManager controls access to the room state, privacy setting, bandwidth, etc.
 - UserManager controls users inside the room, promotes/demotes users, shows a list of users, and accepts or denies users.
 - FileManager give access to the files associated with the room, including publishing new files on the server.
 - StreamManager provides access to the audio and video stream associated with the application.
- *Collaboration User Interface* components allow the developer to reuse the UI for the application and pods.

- *Pods* are high-level, mini applications consisting of (and providing the functionality of) AFCS platform such as web camera, file sharing or whiteboard.

As soon as the authentication processes are completed, the user is allowed to get access to the provided services on the collaboration server. The platform provides six Pods i.e. roster, chat, file sharing, note, web cam, and whiteboard as default services. However, the system developer is able to create or customize Pods to match with the organization's requirements. In this study, we have customized the provided Pods for our virtual conference service. Therefore, all provided services are integrated in one desktop application, as shown in figure V.28. This application is the main module which handles the connection of each Pod.



Figure V.28: User interface of virtual conference modules

The figure above shows screen capture of the virtual conference application. The main window (1) is the first screen that will be displayed to the users when they have logged in to the collaboration server. It provides the chat module which allows cluster members to communicate with text messages. From this window, the user can get access to other services via the control panel (2) at the bottom of the application. The control panel allows the user to enable each Pod from the client side. The Roster (3) button enables a list of online users in the conference room. File sharing (4) makes it possible for users to share their digital files with conference members. The Note module (5) is used for creating the conference note collectively. Then, Webcam window (6) displays multiple video streams from users' web camera. Finally, the Whiteboard module (7) allows the cluster members to express their information/knowledge to the conference members in the graphical view. These Pods were

implemented in our virtual conference service in order to facilitate the collaboration among the cluster members. In the next part, we will focus on the collaboration in peer-to-peer approach.

V.4.2.3. Individual Collaboration

The objective of the individual collaboration service is to transfer information or knowledge between the members in the direct way. There are two major benefits of this type of service to the industry cluster. The first benefit is that it increases the frequency and quality of the collaborations within the cluster. The second benefit is that it allows the experts to share their knowledge to the knowledge users directly via KMS. In order to achieve these, the VoIP server is implemented for improving the functionality of the proposed KMS. The diagram in figure V.29 shows the necessary elements for the implementation.

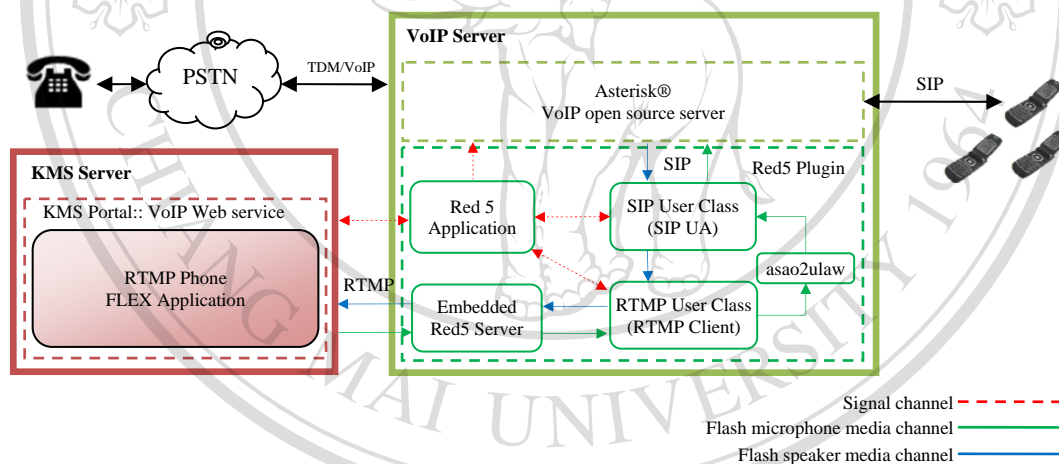


Figure V.29: System architecture of VoIP server and KMS server

The diagram above describes the inter-connection between the KMS server and the VoIP server. The VoIP web service on the KMS server side is developed on Flex technology. The user interface of this application was displayed in figure V.21 (communication). For example, if the cluster leader wants to communicate with one of the cluster committee via this service, he can make a call promptly via the web service on the KMS or the Widget. Then, the request from the KMS server will be encapsulated in the RTMP protocol and send to Red5 Plug-in. This Plug-in is integrated to the system in order to convert RTMP protocol (supported by Flash application) into SIP protocol (supported by VoIP server). Then, the VoIP server will handle the connection between the web service and SIP phone. Integrating

VoIP server not only brings the benefit to the ceramic cluster in term of communication aspect but also extend the capability of the knowledge retrieval of the KMS. An application of this integration will be discussed in the next section.

V.5. Mobile Knowledge Retrieval

In previous sections, we have presented scenarios of collaboration, knowledge sharing, and knowledge retrieval of the industry cluster via the KMS. These scenarios proved that the proposed KMS is able to enhance the collaboration and knowledge of members which will create competitiveness in the industry cluster. Although this system makes it possible for the cluster to create, share, and reuse knowledge liberally (which is the fundamental of the knowledge management theory), there are limitations in terms of place and time. This means that users are able to retrieve their knowledge only if the internet connection is available. Moreover, the web browser application is required for displaying or requesting knowledge. These restrictions sometimes obstruct knowledge workers from acquiring the knowledge from the system when needed. Thus, in order to eliminate such limitations of the proposed KMS, this section will demonstrate an application of knowledge retrieval by combining the advantages of knowledge service and collaboration service. The fundamental of the application is combining the VoIP service with the knowledge query module. The objective of this integration is to support knowledge workers to reuse the knowledge from the knowledge base via their mobile device. Then, the required knowledge is transformed into the appropriate format (text, voice, image, or video) which suits the mobile device. The benefit of the system is allowing the knowledge worker to request or retrieve the required knowledge to immediately solve problems on their task, even in the workplace.

The important technology which allows us to traverse over the barrier of place and time is mobile technology. We cannot deny that the mobile technology is becoming a part of our business in these days. These technologies make the cost of collaboration among the cluster members lower and affordable. Moreover, broadening of internet bandwidth makes VoIP become one of standard protocol for voice communication over the internet in this era. The VoIP application has been integrated in many mobile devices (e.g. SIP phone, soft-phone). In order to enable VoIP communication in the industry cluster, a VoIP server is required for handling the connections from the clients. Nowadays, there are wide ranges of

software and hardware that allow users to establish a VoIP server, even for home-use. In this study, we use Asterisk® [Asterisk 09], which is an open source server for VoIP applications. This server lets us manage the connection of the Session Initiation Protocol (SIP), which is a widely used signaling protocol for multimedia communication sessions, such as voice and video over the internet. Clients are able to connect to their server via their mobile device or desktop application in order to make a connection with other users.

With the intention of exchanging information between VoIP server and KMS server, the VoiceXML [VXML 09] is considered as standard format between both servers. This standard is designed for creating audio dialogs that feature synthesized speech, digitized audio, recognition of key input, etc. Its major goal is to bring the advantages of Web-based development and content delivery to interactive voice response applications. Therefore, the result map on the KMS server has to be re-formatted into VXML format. Then, the VXML content will be interpreted by the voice browser such as VXI* [VXI* 09]. The voice browser module is integrated into the system for reading VXML and converts into text content with the instructions. It also consents to insert scripts and commands into the content in order to support the input from the user. Next, the content and instructions will be transferred to speech synthesis module. This module is the artificial production of human speech. Hence, Text-To-Speech (TTS) software is integrated in the system for rendering normal language text into digital audio format (wave file). Finally, the audio file will be played when the connection to the inquirer is accomplished. Actually, there are many open source TTS modules that are compatible with Asterisk server such as Festival, Flite, Mbrola, etc. In this study, we applied Festival as voice wrapper. Then, Mbrola module was integrated in order to enhance quality of the voice in TTS module. The architecture of VoIP server and its modules are illustrated in figure V.30.

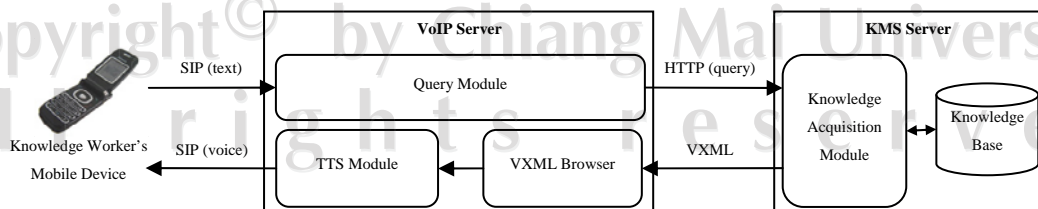


Figure V.30: Knowledge acquisition via VoIP server

The figure above illustrates knowledge acquisition via VoIP server. The main components on VoIP servers are query module, VXML Browser, and TTS module. In order to retrieve the knowledge from the KMS server, the knowledge worker sends the request via a mobile device. The request is the search keyword in text format, which is encapsulated in SIP protocol. Then, VoIP server extracts the search keyword and converts to a query string which complies with KMS's format. Next, VoIP server sends the HTTP request to the KMS server via the knowledge acquisition module. The knowledge map from the inference engine will be converted into VXML standard and sent back to VoIP server. Then, VXML browser and TTS module convert the retrieved knowledge into a voice (speech) file. Lastly, VoIP calls the knowledge worker's mobile device via SIP protocol in order to play the search result in voice form. Without this system, users require a web browser to request and retrieve the knowledge from the system.

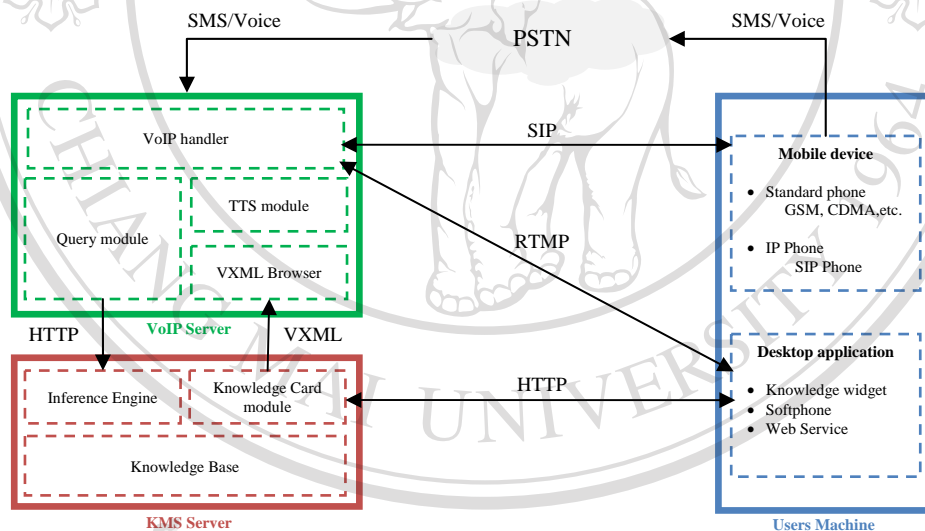


Figure V.31: Interconnection diagram between KMS and VoIP server

The diagram above implies that there are two approaches for reusing the knowledge from the system. The first approach is acquisition via the desktop application. This approach is direct knowledge retrieval from the knowledge system which requires web browser and internet connection. In this study, we create a small desktop application called the *knowledge widget*, for retrieving real-time information from the KMS server. The retrieved knowledge will be transferred back to the knowledge workers via widget. The second approach is acquiring knowledge via the mobile device. In this approach, the request will be transferred

to the KMS server through the VoIP server. Mobile devices can be categorized into two types i.e. standard telephones and IP phone. A “standard telephone” means mobile device that requires PSTN (Public Switched Telephone Network) in order to connect to VoIP server. IP Phone (Internet Protocol Phone) is mobile devices that support internet protocol at their hardware or firmware level. The knowledge request will be send to VoIP server via SIP protocol. Then, the request is encapsulated and forwarded to the KMS server. The search result will be send back to the requester in voice format.

From the communication model in figure V.31, the arrow head indicates the direction of communication with the protocol used for conveying information. The knowledge acquisition process is initiated when the client side sends the inquiry keyword to the server. IP Phone can send a keyword over SIP protocol to the VoIP server. Similarly, a mobile phone is able to send Short Message Service (SMS) through PSTN. An optional way to send a query without a messaging service is to call to VoIP server and press the keypad, following the alphabet layout. The VoIP captures DTMF and converts to a query string. The query module sends the query string to KMS server. However, desktop applications can send query strings to a KMS server directly via web service. The inference engine searches for results from the knowledge cards in the knowledge base. The search results are re-formatted to VXML format and transferred to VoIP server via VXML browser module. The interpreted VXML is converted into digital audio (wave) format by TTS module. Finally, VoIP server makes a phone call and plays search results in audio format to the user. In order to summarize the processes, figure x demonstrates the knowledge acquisition in terms of transferred data. A scenario that demonstrates the knowledge acquisition of the cluster member via a mobile device is illustrated in figure V.32.

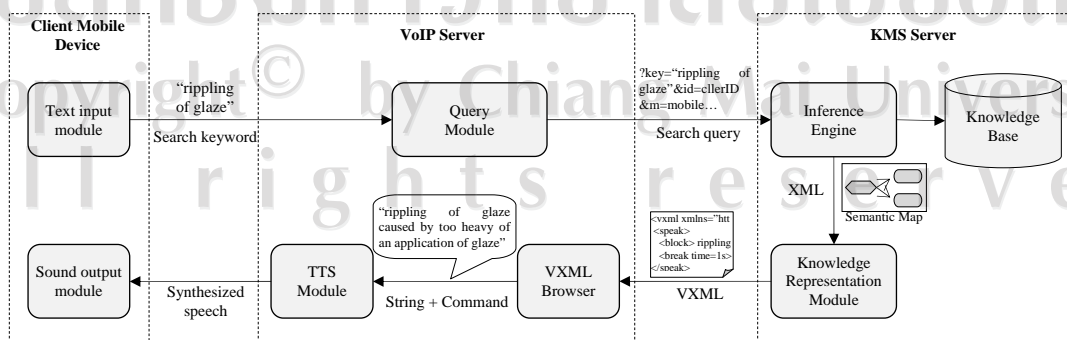


Figure V.32: The knowledge acquisition process in data point of view

The process of this knowledge acquisition will be depicted along with the scenario from the ceramic cluster in our case study. From the example in the diagram above, a knowledge worker on the ceramic glazing process found that the glaze on the ceramic product was rippled. In order to comprehend the cause of this fault, he sent a search keyword (“rippling of glaze”) to the KMS server via his mobile phone. Then, the search keyword was modified into a query string by adding essential parameters for the inference engine. As soon as the inference engine received the query string, it connected to the knowledge base in order to retrieve the knowledge required. The result set from the inference engine (which is in the form of semantic map) is translated into VXML format by the knowledge representation module. (The output from the KMS server in VXML format allows the VXML browser to extract the knowledge content and commands from the script.) Next, the content ids converted into synthesized speech by TTS module. Finally, the knowledge worker receives a phone call from VoIP server. The VoIP handler played the retrieved knowledge “*rippling of glaze caused by too heavy an application of glaze*”. The knowledge obtained from the system helps the knowledge worker to understand the cause of the fault and knowledge related to this fault. Therefore, the problem could be solved immediately. Although there are many advantages from implementing this system in the ceramic cluster, there are some limitations and improving points which will be discussed.

V.6. System Validation

The proposed KMS is applied to a ceramic cluster in Lampang province of Thailand in order to improve the cluster development by focusing on two issues i.e. the knowledge sharing and the collaboration of the cluster members. Our knowledge system is introduced to the cluster committee and CDA for testing. In this process the test specification which was designed in the design level (chapter 4) is used for assessing the functional ability of the proposed KMS. Thus, a scenario of “trade fair preparation” which is one of the key activities of this ceramic cluster; is used for validating our system. This scenario is concern with the BIG+BIH (Bangkok International Gift & Bangkok International Houseware) Fair in 2009. The UML diagram in figure V.33 shows how the cluster members can make use of the provided knowledge and collaboration service on the KMS to achieve their goal in the trade

fair. The sequence diagram comprises two main actors i.e. cluster member and CDA, and two services in the KMS i.e. knowledge service and collaboration service.

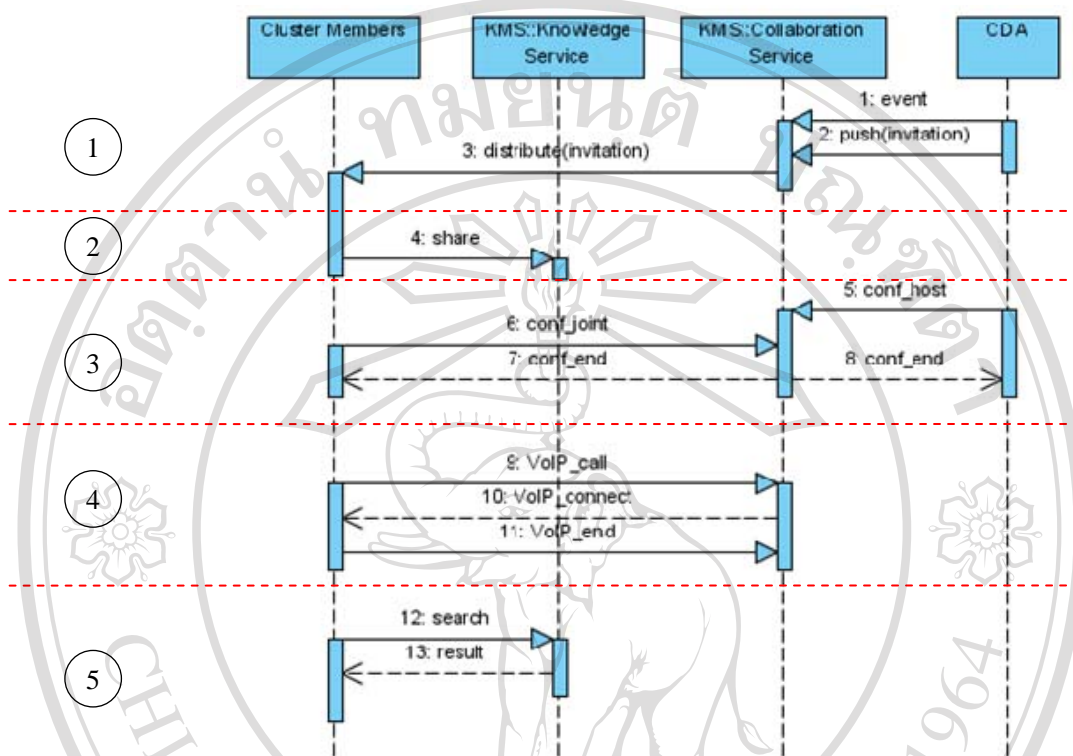


Figure V.33: Sequence diagram of the international trade fair preparation

For this trade fair, this ceramic cluster obtains some subsidization from the government to join this event. Thus, cluster members have to plan their strategy for accessing new markets during this fair. In the past, this cluster did not have any system for supporting these activities. There were only a few meetings before the trade fair which can not satisfy the cluster members. Moreover, the knowledge and experience were not shared among the members. Thus, this scenario will show how the cluster members adopt the KMS for improving this situation, which is divided into five parts as follows.

(a) First part: Mass collaboration

The starting point of this scenario is that the CDA create an event on the c-calendar module in order to inform all members to participate in the virtual meeting via the KMS. Moreover, the invitation is sent to experts in the ceramic cluster by push news module in order to motivate them to share their knowledge about the international trade fair. The

invitation message contains the information about the required knowledge and link to the knowledge card. An example of the push news and mass collaboration model are displayed in the figure V.34.

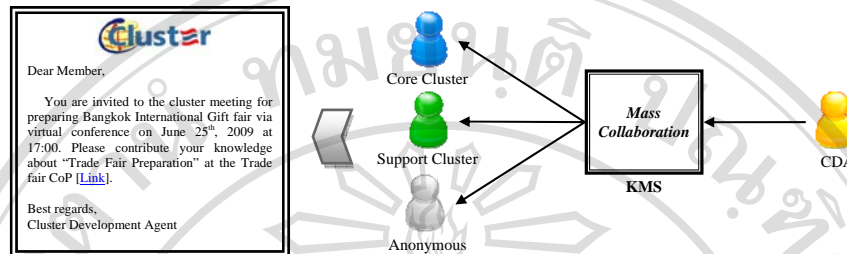


Figure V.34: Mass collaboration model

This type of collaboration is useful for the CDA or the cluster leader to communicate with the cluster members in mass. It also ensures that the information will be distributed throughout the ceramic cluster. After received the message from the CDA, the experts who want to share their knowledge will continue in the second part.

(b) Second part: Knowledge representation and sharing

The experts who are willing to share their knowledge about the international trade fair are able to access the CoP via the provided link in the message. In this part, the knowledge service in the KMS will facilitate the knowledge representation and sharing. The shared knowledge will be stored in form of the knowledge card. An example of knowledge sharing is illustrated in the following figure.

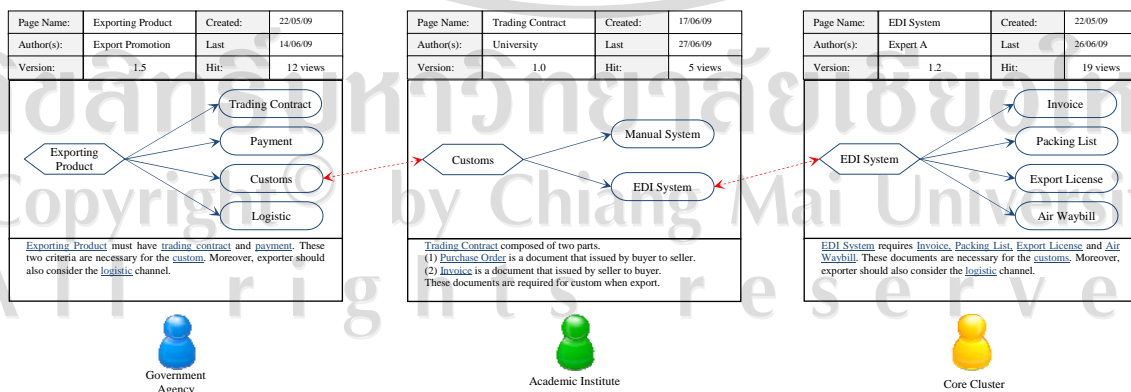


Figure V.35: Knowledge sharing scenario

The figure above demonstrates the knowledge sharing scenario of the experts in the ceramic cluster. For instance, the first expert gets access to the CoP and shares his knowledge about exporting product which is a related task after the international trade fair. Then, the second expert realizes that he has some knowledge about the customs which is necessary for exporting the product. Accordingly, he creates new knowledge card about the trading contract. Finally, the third expert shares his knowledge by extending from the previous knowledge. The knowledge about the EDI system for the customs is added into the system. Then, each knowledge card will be linked together automatically by the inference engine. With this concept, the new knowledge could be added to the knowledge base liberally.

(c) Third part: Group collaboration

In the meeting, the virtual conference module is initialized by the CDA who logged in as the host of the meeting room. Then, the cluster members and anonymous users can join the meeting room with the different privileges. The model of the group collaboration via the KMS is illustrated in the figure V.36.

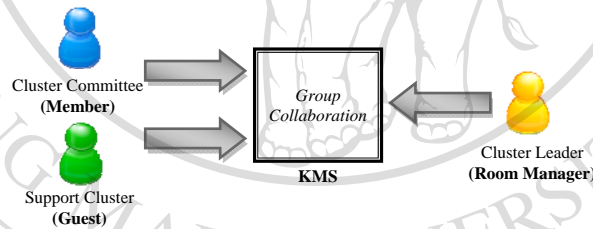


Figure V.36: Group collaboration model

The figure implies that there are three types of privilege on the KMS i.e. room manager (host), member (participant), and guest (audience). The benefit of the virtual conference module allows cluster members to organize the meeting from the distance. The knowledge and information about the trade fair preparation is shared in this stage.

(d) Forth part: Individual collaboration

After cluster leader assign the task to the member from the meeting in the previous stage, the individual collaboration service is considered for supporting the communication exterior the meeting. This VoIP module is not only allows the cluster members to communicate via the VoIP web service on the knowledge widget but also via their mobile

devices. This will support the communication among the members of the cluster as shows in the following model.



Figure V.37: Individual collaboration model

The model above shows that the VoIP module supports the communication between the members in peer-to-peer approach. The details information which is unable to be shared in the meeting will be shared in this stage. The cluster members exchange their information (e.g. progress of preparation, urgent problems, etc.) before the next meeting via this service.

(e) Fifth part: Knowledge retrieval

In this stage, the search module in the knowledge service is responsible for the knowledge retrieval. The knowledge user is able to send a request for searching the required knowledge from the cluster knowledge base. The knowledge retrieval model of our proposed KMS is showed in the figure V.38.

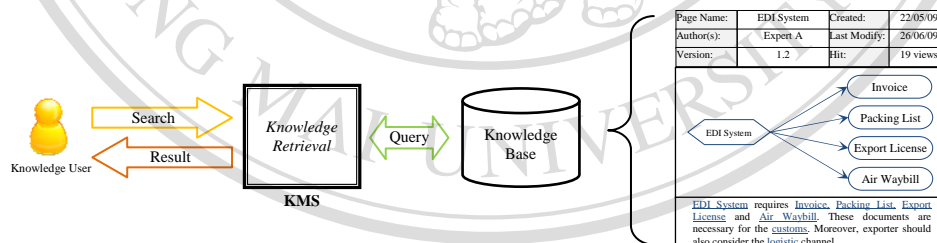


Figure V.38: Knowledge retrieval via KMS

The model above shows the process of knowledge retrieval from the KMS. For example, a knowledge user requires the knowledge about “EDI System” from the KMS. The keyword is sent to the search module. Then, the inference engine browses over the knowledge base in order to query the knowledge cards which match with the keyword. Finally, the search result in knowledge card format is delivered to the knowledge user in an appropriate form.

V.7. Conclusion

In summary, this chapter showed how the design model is transformed into the KMS. Each tool in the system architecture is selected with regard to the cluster's requirements in both knowledge and collaboration aspects. Concerning these requirements the knowledge service is introduced to the system architecture for managing the knowledge activities of the cluster members (i.e. representing, sharing, and reusing). Two major modules that facilitate these activities are knowledge card and inference engine. The notion of knowledge card was adopted for representing and sharing the knowledge of experts in the cluster. On the other hand, the inference engine was designed for searching the knowledge through the semantic knowledge model in the knowledge card. These services make it possible for cluster members to share their knowledge through an indirect approach and still retain the quality of knowledge.

Collaboration service is supported by collaboration and communication tools (such as c-calendar, push/pull news, virtual conference, etc.) with the purpose of enhancing the coverage of information and degree of collaboration. In the knowledge management point of view, the major categories of exchanged knowledge on this level are know-who, know-where, and know-when. Cluster members gain knowledge about other persons from the cluster map, and opportunities from calendar and push/pull news. However, deeper level of knowledge (e.g. know-how, know-what, know-with) can be obtained via the virtual conference or the VoIP service. The obtained result from our investigation (presented in chapter 4) confirmed that cluster members feel uneasy to request/share their in-depth knowledge in the direct approach. They preferred to exchange their knowledge via CDA. Therefore, this situation indicated that CDA is an important element for the industry cluster development. Although, this method is able to facilitate the knowledge sharing in the cluster, the quality of knowledge is decreased and the knowledge could not be reused in the future request.

The last group of service which was developed for the system is the presentation service. The main objective of this group is to present the provided services on the KMS to the users in the appropriate form. Many studies [Pentland 95] [Schreiber 99] [Malhotra 04] consented that one of the failures of knowledge system development is neglecting specific characteristic of the organization. Thus, this presentation services aimed at introducing the

KMS to the organization in regard to cluster's preference. It allows cluster members personalizing the KMS to suit with their role. In addition, it provides the possibility for the users to interface with KMS in three ways i.e. server-side application, client-side application, and mobile device. In order to support this, KMS portal, knowledge widget, and VoIP services are implemented in the system to increase the usability of the KMS. At the end of the chapter, three scenarios of KMS usage were demonstrated in order to verify the application of the KMS in the industry cluster.



ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่

Copyright© by Chiang Mai University

All rights reserved

Conclusion and Perspectives

The ultimate goal of this research is to improve the competitive advantage of the industry cluster by focusing on the collaboration and sharing of knowledge among the members which are the key success factors of the industry cluster. Therefore, our study is organized into five parts (chapters). Each chapter focused on different point of views of the study. The explicit results from each chapter are described as follows.

The **first chapter** aims at analyzing the industry cluster context and the need for knowledge management in order to improve the competitiveness of the cluster. We found that the innovation is a driving force behind the long-term competitive advantage of the industry cluster. Moreover, in order to develop the innovation of the cluster, knowledge and collaboration are the foundation of the development. The literature review related to the clusters development in many countries also confirmed that these elements are the key success factors of the cluster development. However, we found that the main obstacle of the development is the “co-petition” relationship. This type of relationship in the industry cluster context is more intense than supply chain, extended enterprise or other organization networks. Thus, the critical problems of this study are set to understand the characteristics of information and knowledge sharing of the cluster, and to propose the architecture to improve this situation.

The **second chapter** concentrates on the application of the organizational knowledge management methodology to the cluster context. We found that among the cases that have been studied; none of them relied on the knowledge management approach. Thus, we propose the KMS for facilitating and improving the industry cluster development. The knowledge engineering methodology is adopted for supporting analyzing, designing, implementing the KMS for a specific context. This chapter compares various approaches of knowledge engineering methodology (i.e. MOKA, SPEDE, AKEM, and CommonKADS) in order to select an appropriate method for to be applied in the industry cluster context. The result of the analysis shows that CommonKADS methodology is the most suitable for our case study.

However, it cannot be directly employed without a certain adaptation. Thus, the **third chapter** concentrates on adapting the methodology to suit with our context. In our methodology, the “*model suite*” is proposed to design the KMS. It is composed of four levels i.e. context, concept, design, and implementation. The context level is proposed to analyze the organizational context of the cluster in terms of actors, processes, knowledge-intensive tasks, and knowledge assets. The concept level is defined for extracting the knowledge from the experts and also collaboration model of the cluster. The design level was added to the model suite of convert the cluster’s requirements into a system specification by using software engineering approach. Finally, the implementation level is oriented to the KMS development process.

The **fourth chapter** shows the obtained results from the application of the model suite to a ceramic cluster in Thailand. This chapter concentrates mainly on the first three levels of the model suite. The results from the context level are composed of three models i.e. cluster, organization, and task. The cluster model implies the physical network of the ceramic cluster. The organization model reveals the present organizational context of the ceramic cluster e.g. level of trust, structure, and knowledge sharing. The task model focuses on selecting the knowledge-intensive tasks and knowledge assets for our case study. In this level, we selected a task about “*finding market opportunity*” as our instance. The cluster committee has consensus that this task is the most significant task in the ceramic cluster in the present situation. The concept level is composed of two models i.e. knowledge model and collaboration model. The knowledge model is used for extracting the expert’s knowledge about the specified task and then converts it into semantic knowledge map. The collaboration model concentrates on the approach for exchanging the information within the cluster. Finally, the design level interprets the requirements from the previous levels into system scenarios, specifications, and architecture. The outcome from this chapter is the specification documents to develop the KMS. These documents are also considered as the medium between the knowledge engineers and the system developers.

The **fifth chapter** concentrates on the development and the implementation of KMS for the specific case of ceramic cluster. The KMS architecture is separated into three levels of services: collaboration, knowledge, and presentation. Each level aims

at supporting the industry cluster in the different aspects. The collaboration services comprise the collaborative tools such as push/pull news, collaborative calendar, and virtual conference. The objective of these tools is to improve the collaboration of the cluster in terms of frequency and quality of collaboration. The knowledge services are supported by two main modules i.e. the knowledge card and the inference engine. The knowledge card module is developed in this study in order to facilitating knowledge creation and sharing of the experts in the cluster. The advantages of this concept are related to the possibility for experts to express their knowledge in different ways (semantic map or wiki) and the possibility to append their knowledge to other's knowledge. The main technology behind these benefits is the semantic map module. It provides many advantages to our knowledge system e.g. readability by both human and machine, simplicity to manipulate the knowledge, compatibility with the inference engine, which supports the users to reuse the knowledge in the system. It provides better mining results than the ordinary content search due to the result set generated by semantic relationship of the concept (in the semantic map module).

The implementation scenarios are also presented in this chapter in order to depict how the KMS can improve the situation of the ceramic cluster. The first scenario shows the collaboration possibilities via the KMS. It proved that our knowledge system supports the three types of communication (i.e. peer-to-peer, group, and mass). The second scenario demonstrates the possibilities in term of knowledge creation, sharing and retrieving via the KMS. This scenario shows that major knowledge activities of the cluster members are supported by the proposed architecture. Finally, the last scenario shows the integration between the knowledge service and the collaboration service in order to eliminate the barriers of knowledge retrieval in term of place and time. In this scenario, the mobile device is used as the tools for acquiring the knowledge from the server. This allows the users to reuse the knowledge from anywhere and anytime, even their workplace (production lines, logistic sites, exhibition halls, etc.).

From our **perspective**, although this study showed the ability of the KMS for supporting the industry cluster development, it can be improved in two aspects: Methodological aspect and Technical aspect. The methodological improvement concerns the knowledge engineering methodology which is the principle of this study.

The proposed methodology is based on CommonKADS methodology which is considered as more suitable for applying in the industry cluster context than the others (MOKA, SPEDE, and AKEM). However, it requires an enhancement in term of knowledge elicitation (knowledge modeling) process. The provided knowledge model in CommonKADS is subjective to the engineering domain which makes uneasy to KE to apply the proposed methodology in the soft domain. Thus, enhancing this part of our methodology will help in generalize the methodology into various domains.

The validation of the methodology in the medium-term is one of the critical points in our perspective. The methodology validation has been done in our study only in event-based aspect (short-term) due to the limit of time. However, the evolution of the KMS during the first business activity cycle of the cluster (medium-term) is highly important for the knowledge management project. Thus, the CDA is required to motivate and facilitate the cluster member to collaborate and share their knowledge via the system. The study of evolution of KMS and the cluster behavior in medium-term will help us to improve the methodology.

In the technical point of view, the KMS architecture itself can be improved in two points. The first point is the collaboration services which are the fundamental of the information and knowledge exchange of the proposed architecture. The architecture of these services can be enhanced by the concept of Service-Oriented Architecture (SOA) which brings ability to the KMS to exchange data with other systems. Another point is the knowledge services level which allows experts to create, represent, and share their knowledge. The concept of ontology can be applied in the Collaborative Knowledge Card system in order to formalize the knowledge map into the structured form. Moreover, this concept make possible for the knowledge engineer to maintain the knowledge map in the KMS.

The last perspective is the extensibility of the methodology and the platform into other industry clusters in the different domains. We are interested in generalizing our methodology to the domain of tourism cluster which seem to have similar characteristics as the ceramic cluster. The proposed KMS would help cluster members to exchange their knowledge and collaborate on the tourism business activities.