Chapter III

Proposed Methodology

III.1. Introduction

Although there is ambiguity about methodology for the industry cluster development, the fundamentals of the development are consensual. According to Rosenfeld [Rosenfeld 97], *"cluster development can be done in various ways, depending on characteristics of the cluster and propriety. Participants gain benefits and advantages from being members of the cluster through collaboration"*. The study of DTI [DTI 05] confirmed this statement that *"the critical factor for successful cluster development is the formal and informal flows of knowledge that is generated from the network"*. Accessing the knowledge can support collective learning and enhance the competitive performance of the industry cluster. These factors imply that the foundation of cluster development is managing the knowledge and collaboration of cluster members. Hence, the purpose of our methodology is aimed at applying the knowledge management practice to the industry cluster development context.

Actually, the transferring of the knowledge through networks and partnerships can be achieved by two methods, direct and indirect sharing. Direct sharing (e.g. face-to-face contact or synchronous communication) allows cluster members to transfer their knowledge to the knowledge user accurately. Although direct sharing seems to be the most effective method for transferring the knowledge, there are many limitations in this method (e.g. place, time, characteristic of relationship, level of trust, etc.). On the other hand, indirect sharing (e.g. a synchronous communication or sharing knowledge via an information system) is a less effective method in terms of quality of knowledge. However, this method enables knowledge sharing, while direct sharing is impracticable. As we described the "*co-opetition*" relationship among the cluster members, this makesthe enterprises which are competitors feel uneasy to share their knowledge in the direct way. This situation obstructs knowledge sharing, which will decelerate the cluster development. Hence, the indirect method was used to overcome this obstacle. Cluster practitioners used the advantages of technology, such as interactive cluster portals, to facilitate networking, and share information about the cluster as well as using them for actual business to business interactions [DTI 05]. The Knowledge Management System (KMS) is considered as a solution for enabling the knowledge flow and managing the knowledge of the industry cluster.

The KMS brings various benefits to the industry cluster. It enables the members to have ready access to the knowledge base of the cluster. It supports the knowledge management activities e.g. creation, representation, sharing and retrieval of the knowledge. KMS can also reduce the limits of collaboration among members by using the advantages of information technology. Although there are some studies about the industry cluster in the knowledge context [Bornemann 03] [Levy 03] [Malmberg 04], the methodology of the knowledge management (or KMS) for the industry cluster currently does not exist. Although there are many information systems which are claimed as organizational knowledge management systems, they could not be directly applied to the industry cluster context due to several factors:

1 The differences in the type of relationships within a single industry cluster such as collaborator, competitor and the shared-resource relationship. This separates sharing of knowledge in the cluster into many levels regarding trust. Most organizational knowledge management systems do not address this matter.

2 An industry cluster is a loose network, unlike a supply chain. The knowledge shared among the members is dependent on the strengths, weaknesses, opportunities and threats in each cluster. It does not only depend on the product itself, but also the business environment. For this reason, the process-based knowledge management system has failed in the industry cluster; even though it has had great success in the manufacturing context.

3 The industry cluster includes a variety of groups of users (e.g. enterprises, government agents, academic institutes, supporting industries, etc.) which take different roles in the cluster. Each group offers and requires a different type of knowledge from the system. Developing a KMS for the industry cluster thus requires a prudential analysis to suit each cluster.

The KMS architectures which were proposed in chapter 2 revealed that there is no exact architecture. The specification of KMS depends on the characteristics of the organization. Thus, in this chapter, we will propose the methodology for developing a KMS for the cluster with the Knowledge Engineering (KE) method. In the study, we adopted the CommonKADS which is knowledge engineering methodology for designing a KMS for the industry cluster. Knowledge engineering methodology helps us to analyze the requirements of cluster members. Moreover, it places strong emphasis on the conceptual modeling of the knowledge-intensive activities. The output of the knowledge engineering methodology provides us the feasibility analysis, knowledge models and specifications of the KMS.

CommonKADS was used for developing the knowledge management system in various domains. For examples, it was applied to the financial domain for developing mortgage assessment systems [Schreiber 99]; to the manufacturing domain for improving product manufacturing processes and to the automation domain for developing mobile robot control. Nevertheless, the CommonKADS methodology itself could not be straightforwardly deployed in the industry cluster context, due to many reasons which will now be described. Thus, the following part depicts why and how we adapted the CommonKADS methodology within the industry cluster context. Afterwards, the research framework will give details of each level of the proposed methodology, with examples. Finally, the assimilation of the methodology will be depicted.

III.2. Adapting CommonKADS to the industry cluster context

CommonKADS provides a "*CommonKADS Model Suite*" as a core of the methodology. The model suite was developed from a need to build industry quality knowledge-based systems on a large scale in a structured, controllable and repeatable way. The CommonKADS model suite consists of three groups of models, each with its own focus, due to the fact that there are several different aspects that need to be investigated. Although CommonKADS became the *European de facto standard* for knowledge analysis and knowledge-intensive system development, it was developed for well defined, formalized, "hard" domain(s) [Sandberg 96]. However, the industry cluster development is loosely defined and does not provide well-established theories or methods. From our investigation, although CommonKADS appears to be the most suitable methodology for the industry cluster which is a “semi-soft” domain, some modifications in its model suite are required to match with the organizational context and environment.

The CommonKADS model suite provides six model templates as we described in the previous chapter, which highlight the different aspects of the design space i.e. organization, task, agent, knowledge, communication, and design model. However, in the industry cluster context, some CommonKADS models could not be promptly applied. Thus, in this part we will describe *why* and *how* we customized the CommonKADS methodology in the industry cluster context. From our initial investigation into the industry cluster, it is clear that there are some difficulties in applying *agent model* and *communication model* in the industry cluster context. Thus, we have replaced these models with appropriate models that will be proposed in the following section. Moreover, the *design model* of the CommonKADS methodology is not defined in a concrete manner. This might lead to misinforming requirements and specifications from the knowledge engineer to the system developer. Therefore, we have modified the design model in order to emphasize and clarify the output from the methodology. Hence, the modification of the CommonKADS models is shown in figure III.1. The dashed outline boxes represent the models which required the modification or replacement.

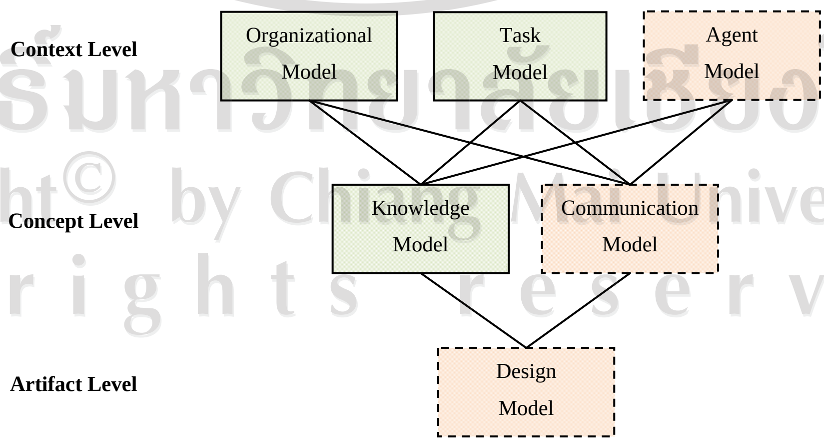

Figure III.1: CommonKADS model suite

The **agent model** involves collecting lists of executors of a task and describing the characteristics of the agent. The objectives of this model are specifying the list of tasks that are performed by the agent, lists of other agents communicated with, the knowledge items possessed by the agent, competence of the agents, and responsibilities and constraints. However, in the industry cluster context, agents (known as cluster members) were not well-defined, which differs from usual organizations. Thus, we adapted a method called cluster mapping for extracting the list of agents from the industry cluster. So, the agent model would be replaced by the *cluster model* which will be briefly described in the proposed methodology section.

The **communication model** focuses on modeling the communicative transactions between the agents involved. A transaction tells what information objects are exchanged between what agents and what tasks. Thus, the objective of this model is structuring the information exchanging of the organization. However, the characteristics of communication in the industry cluster also differ from the intra-organizational communication. Due to the structure of cluster organization being flat, communication among the members is difficult to structure. Besides, the behavior of the industry cluster in exchanging information is more complex. Thus, the communication model would instead be replaced with the *collaboration model* in order to analyze the characteristic of collaboration in the industry cluster. In order to represent the communication model of the industry cluster, the scenario model was used to analyze behavior of communication in the cluster.

In order to develop an accurate KMS for the industry cluster, the **design model** takes an important role for interpreting requirements into the system specification. This model not only allows knowledge engineers to verify the system with cluster members, but also to convey the blue-print of the system to system developers. Unfortunately, this model was lightly defined in the CommonKADS methodology. Moreover, the output of the design model relies on some specific software tools. Therefore, we have intensified and generalized the design level with the software engineering concept, which comprises three modules, system specification, scenario and architecture. The outputs from these modules are system specification documents which are ready to hand over to system developers for the developing process.

Hence, this study has proposed a new methodology for designing the knowledge system for the industry cluster by adapting from the concept of CommonKADS methodology, as shown in figure III.2. The model suite combines the cluster analyzing technique; CommonKADS knowledge engineering methodology and software engineering concept together, in order to support knowledge engineers to analyze the organizational context, capture the knowledge concept, and design the knowledge system for the industry cluster. The proposed model suite was divided into four levels: context, concept, design, and implementation level.

The model suite presents our knowledge engineering methodology, which is a core of our framework for designing and implementing the knowledge management system for the industry cluster. The thickly outlined boxes indicate the modules modified from the original methodology. The advantage of this framework is that it does not concern only the ‘hard’ side (e.g. structure, process, task, etc.) but also culture, behavior, and opportunity which are the ‘soft’ side. The hierarchic and complementary nature of the models in terms of model-refinement and viewpoints, functions particularly well for soft domain. This is because it provides complementary descriptions of the acquired or constructed aspects of a knowledge-based system, which gives a good handle on consistency and completeness checking [Sandberg 96].

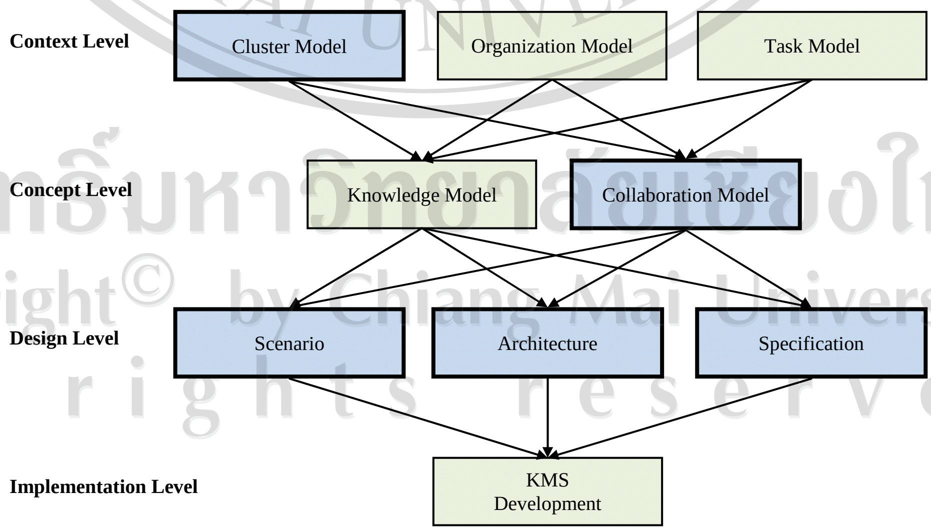

Figure III.2: Proposed methodology for developing KMS for the industry cluster

In the following sections, we will explain how to conduct the knowledge system development project within this framework. The details, techniques and tools of each model in this research framework will also be briefly described. Finally, the integration of these models in terms of input and output will be clarified.

III.3. Research Framework

A knowledge-based system must be managed by learning from the experiences in a controlled "*spiral*" way, because knowledge is too rich and too difficult to understand to fit into a rigid approach [Schreiber 99]. The spiral method is repeating and revising the different steps in the cycle of development. The spiral model approach was originally devised by Barry W. Boehm [Boehm 88]. This model of development unites the features of the prototyping model with an iterative approach of system development, combining elements of design and prototyping-in-stages. The spiral model represents the evolutionary approach of IT project system development and carries the same activities over a number of cycles in order to elucidate system requirements and its solutions. This model attempts to combine the good features of the prototyping model and the waterfall model approaches. The prime difference between the waterfall model and the spiral model is that the project system development cycle moves towards eventual completion in both the models, but in the spiral model the cycles go back several times over earlier stages, in a repetitive sequence. The spiral model approach was illustrated in figure III.3.

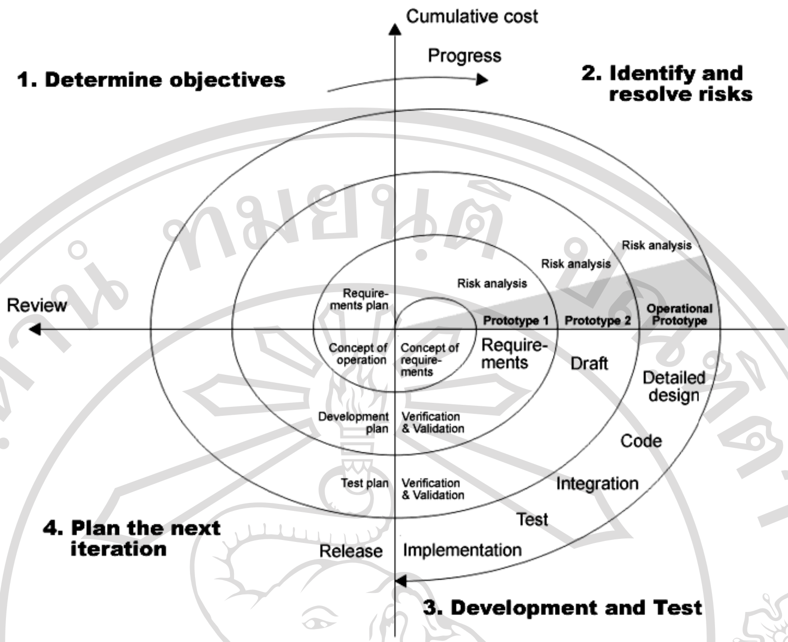

Figure III.3: Spiral model approach [Boehm 88]

The four quadrants indicate recurring and structured steps of project management activity. The project management cycle consists of four activities (determine objectives, identify and resolve risk, development and test, and plan the next iteration) that recur in every cycle of the project. Accordingly, the model suite in this study also requires the repeating and revising of the levels of methodology. The context level generates organizational requirements for the knowledge management project. Then, the concept level produces the detailed requirement of the system. Afterward, the design level converts requirements into detailed design. Finally, implementation level develops and tests the system before releasing the KMS. However, feedback from users is necessary for revising and improving the knowledge system to suit the organization. The comparison of the spiral model approach with our research framework is illustrated in figure III.4.

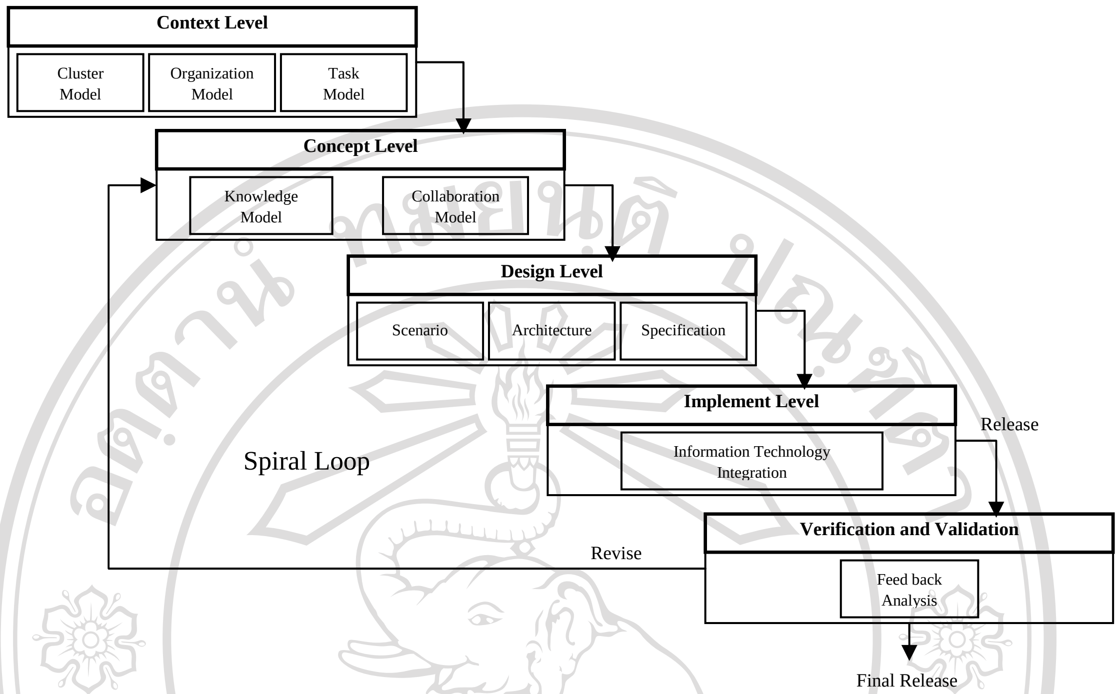

Figure III.4: The proposed methodology in the spiral approach

The diagram above shows the procedures of conducting the knowledge management project. The spiral loop enables us to improve the knowledge system from the users' feedback. However, it is not necessary to repeat the context level again, due to it giving only the concept of the requirements and the fact that it already has consensus from the members of the organization. On the other hand, the concept level contributes the detailed requirements i.e. the knowledge model and collaboration model for the system.

In this section, the details of elements in each phase will be described, while the results from applying the proposed methodology in the industry cluster will be demonstrated in Chapter 4. Then, output from the design level (requirements and specifications from the cluster) will be used to create detailed design and integration of information technology for KMS, which will be presented in Chapter 5.

III.3.1. Context Level

Context level is comparable to the initial investigation phase of the spiral model. This phase is very important for the project management. It introduces better understanding and deals properly with the wider organization context. This level supports not only the knowledge engineer to comprehend the environment of the organization dealt with, but also knowledge managers, knowledge users and experts to achieve consensus for the required knowledge in their organization. Many failures in knowledge system development have resulted, not from problems with the technology itself, but from the lack of concern for social and organizational factors [Schreiber 99]. CommonKADS methodology called this phase "*knowledge orientation*" the goals of which at this level can be explicated as follows:

1 *Identify problems and opportunities* which knowledge management system can provide added value to the organization, and the risk in the project.

2 *Decide about solutions and feasibility* of the project in terms of expected cost, benefits, technological feasibility, and needed resources and commitments within the organization.

3 *Improve task and task-related knowledge* by analyzing the required knowledge of the task in a selected business process, and what improvements can be achieved in this respect.

4 *Plan for organizational changes needed* by investigating the impacts from deploying knowledge management system to the organization.

The context level comprises 3 models: cluster model, organization model, and task model. Each model concerns a different organizational context in order to achieve the knowledge management project. It provides the concept of requirement for the knowledge engineer, which will be used as the input for the concept level. Thus, it is very important to follow the order of the model in this framework.

III.3.1.1. Cluster Model

The cluster model was proposed to replace the agent model of CommonKADS for analyzing the participants in the industry cluster. Identifying the network, collaboration, potential and strategy of the industry cluster are important tasks which have to be done before developing a cluster. Thus, there are many qualitative and quantitative methods that support this model. Porter [Porter 98a] recommended the cluster map and the diamond model as fundamental tools for the initial analysis of the industry cluster. The cluster map aims at visualizing the participants in the industry cluster in the layout of a map (such as group or geography), while the diamond model aims at analyzing the economical environment of the cluster. Both tools are used for initial investigation in most of the cluster development projects.

(a) Cluster Mapping

The cluster map was originally adapted from the Social Network Analysis (SNA) theory. It was popularly used for mapping and measuring of relationship and flow between people, groups, organizations, computers, web sites, and other information/knowledge processing entities. However, these techniques are depending on the objective of the analysis and required outputs. To understand the industry cluster network, we evaluated the *location of participants* in the network. Measuring the network location involves finding position of nodes in the network. These measurements give us insight into the various roles and groupings in a network.

From the given definition in the first chapter about the industry cluster, the cluster map can be separated into 5 main groups of the members in the cluster: core cluster, government agencies and associations, financial and academic institutions, supporting industries, and downstream industries. The position of each group is displayed in figure III.5.

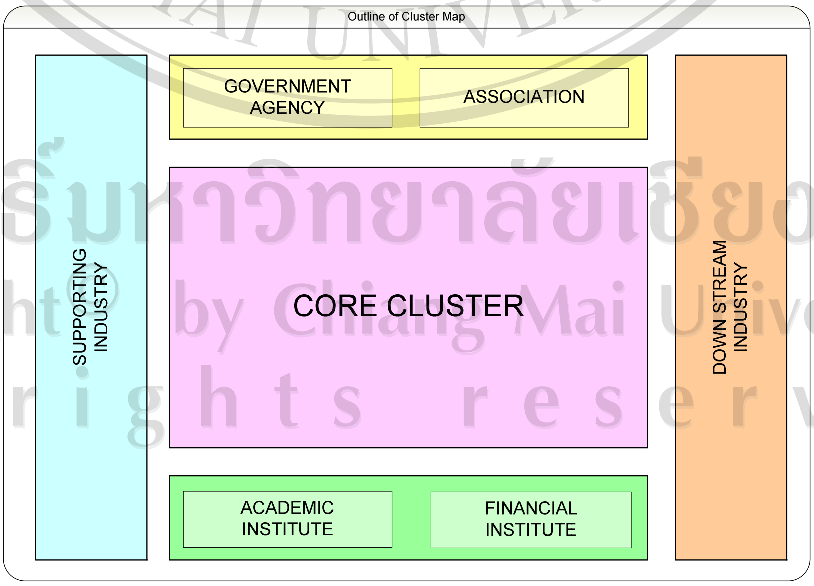

Figure III.5: The outline of the cluster map

In order to extract a cluster map from the industry cluster network, one of the knowledge elicitation techniques, called 'structured interview', has been applied. The structured interviewing comprises a set of questions. *The "10 Questions Technique"*, proposed by UNIDO [Dawson 03], was used to analyze the location of participants in the industry cluster network. The set of questions are listed below:

Q1. What companies do similar business to yours?
Q2. What companies produce and supply raw materials for you?
Q3. What companies provide these services for you: logistics, warehousing, maintenance, financial, training, consulting, advertising, product distributing, marketing and communications? How importantly do those companies affect your competitiveness?
Q4. What companies sell equipment for production for (to?) you?
Q5. What academic institutions provide manpower meeting your company's requirements?
Q6. What institutions support you in terms of research and development?
Q7. Of what associations or specific institutions are you a member?
Q8. What government offices are involved with your company the most?
Q9. How does a government department affect/influence your ability to be competitive?
Q10. Does your company need financial support from financial institutions? If so, what kind of finance source do you expect from them?

One result from interviewing is a list of the participants involved in the industry cluster, which are categorized into five main groups: core cluster (Q1), supporting industry (Q2-Q3), downstream industry (Q4), academic institute & financial institutes (Q5-Q7), and association and government agencies (Q8-Q10). The number of inquiries repeated depends on the quality and quantity of the answers obtained. However, it is not necessary to complete questioning of all the participants in the cluster map in the beginning of the analysis. New entities can be added to the cluster map later to update it. The results from each inquiry are combined by using union (U) operation as shown in the equation below.

$$M = m_1 \cup m_2 \cup m_3 \cup m_4 \ldots \cup m_n$$

Where ***M*** is combined map and ***m*** is obtained map from each interview

To illustrate this, figure III.6 (left-hand side) shows 4 interviews with different organizations in the handicraft cluster (i.e. m_1, m_2, m_3 and m_4). The results from each interview are combined and converted into the cluster map as shown on the right-hand side (*M*).

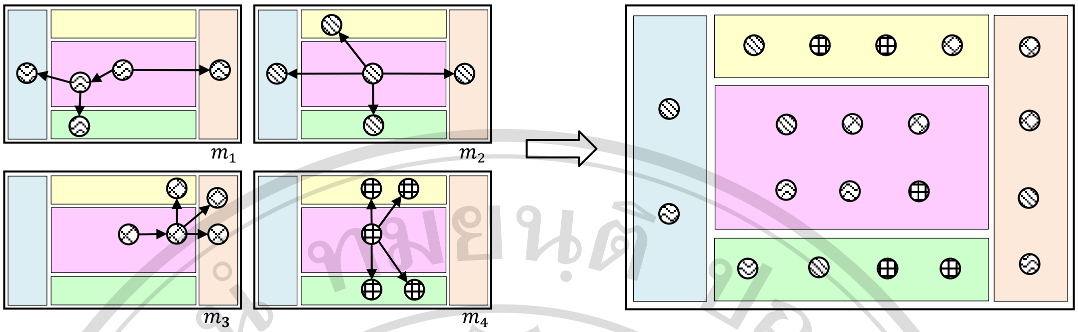

Figure III.6: Cluster maps combining

The cluster map above represents two types of information. Firstly, the arrows in the cluster map represent the inter-connection of the cluster members. This information is useful for doing the social network analysis. Secondly, the group of organizations in the cluster map represents the role of each member in the cluster. This type of information was used for understanding the responsibility of the organization to cluster. This map is mostly used for representing the physical model of the industry cluster. The list of grouped members in this map will be referred by other models. Another technique usually used for analyzing environment of the industry cluster is the diamond model, which will be presented in the following part.

(b) Diamond Model Analysis

The "Diamond Model" [Porter 90] is a well-known technique which is used for analyzing the economical environment of the industry cluster (e.g. potential, strategy, chance, etc.) as showed in figure I.6. The model consists of 5 main elements: government, factor conditions, demand conditions, related and supporting industries, and firm strategy, structure and rivalry. Table III.1 shows the list of criteria of the diamond model with descriptions from the handicraft cluster context.

Criteria	Analysis
Government	*Factors from government*
Firm strategy, structure and rivalry	*Factors from competition in the industry*
Factor condition	*Factors from condition of the industry*
Related and supported industries	*Factors from related industry*
Demand condition	*Factors from the demand-side*

Table III.1: The outline of Porter's diamond model

The interpretation of the diamond model provides the macro-economical view of the handicraft cluster. Information from this model is mostly concerns the government or cluster development organization in order to give suitable support for the industry cluster. However, considering the macro-view of the industry also helps the knowledge engineer to identify external factors of the industry cluster which may affect the knowledge management project. Due to diamond model analysis being fundamental to the industry cluster development, there was much secondary data from earlier studies. For example, in Thailand, there was a study supported by the Thai government to analyze all potential clusters in Thailand [NESDB 04] and diamond model was one of the tools used in that analysis.

III.3.1.2. Organization Model

The organization model aims at investigating and explaining the problems and opportunities, general context, and potential solutions that the introduction of the KMS would bring. This model also includes functionality for investigating organizational structure, processes, impacts and feasibility. The CommonKADS methodology provides five worksheets (OM-1 to OM-5) for analyzing the organizational environment and the corresponding critical success factors for the knowledge system. An overview of the organization model is illustrated in figure III.7.

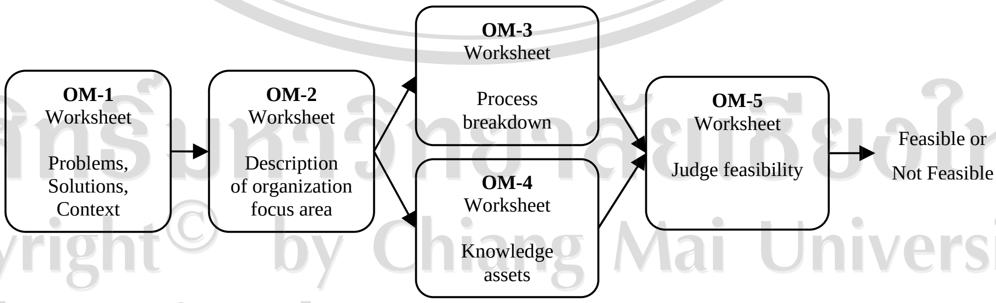

Figure III.7: Organization model processes and worksheets

The worksheets OM-1 to OM-5 are used for interviewing knowledge decision makers in organizations. Then, the outputs from the model are the list of the *knowledge intensive tasks*, *assets* and *agents* which are related to each task. Finally, the feasibility of the knowledge management project was analyzed to see if the project was feasible in terms of business, technique, project and solution. It serves as a decision support for an economical, technical and project feasibility study, in order to select the most promising focus area and target solution. The details of each worksheet are presented as follows:

The OM-1 worksheet investigates the circumstance of the industry cluster from a micro economical point of view. It explains the various aspects to consider, and helps in specifying the organization. In this study, this worksheet was used for interviewing key staff members such as the cluster leader, CDA, or responsible government agency. In addition, some information in this worksheet could be obtained from the diamond model.

Organizational model (OM-1)	Problems and Opportunities
Problems and opportunities	*Problems:* *Opportunities:*
Organizational context	*Vision:* *Missions:* *Strategies:* *Supply chain:*
Solution	*Proposed solutions:*

Table III.2: The outline of Organization Model worksheet 1 (OM-1)

The OM-2 worksheet is derived from OM-1 but concentrates upon the focused area of the organization. The worksheets contains information regarding the structure of the organization, the process in focus, the people involved, the type of resources used, the knowledge (as a needed resource used within the process in focus), and the culture and power which pays attention to the different “invisible” factors such as unwritten rules. In the general view, this worksheet was extended from OM-1 but focused on the in-depth analysis within the organization. This worksheet could be used for interviews with the same group as OM-1.

Organizational model (OM-2)	Variant Aspect
Structure	*Organization structure*
Process	*Processes, tasks, activities*
People	*Members, responsible, stakeholders*
Resource	*Information system, equipment, technology*
Knowledge	*Knowledge in organization*
Culture and power	*Unwritten rules, social context*

Table III.3: The outline of Organization Model worksheet 2 (OM-2)

The OM-3 worksheet provides a break-down of the processes specified in the OM-2 worksheet. It indicates what the knowledge intensive process is and what knowledge is used. The document also provides us with information on the knowledge worker who is performing the task, list of resources used and how significant the task is. This worksheet was used for interviewing experts or process managers in each process declared in OM-2.

Organizational model (OM-3)		**Process Breakdown**				
No.	**Task**	**Performed By**	**Where**	**Knowledge Asset**	**Intensive**	**Significance**
no.	*task name (from process in OM-2)*	*People, system (from people and resource in OM-2)*	*Location (from structure in OM-2)*	*List of knowledge resources*	*Boolean (yes/no)*	*Significance of the knowledge*
…	…	…	…	…	…	…

Table III.4: The outline of Organization Model worksheet 3 (OM-3)

The OM-4 worksheet is broken-down and investigates the knowledge assets that were identified in OM-3. Along with the aid of the previous worksheet, it specifies the person who is the owner, what knowledge assets are used, and where it is used. OM-4 also provides information regarding if the knowledge used is in the right form, right place, right time and right quality. This worksheet was used for interviewing the experts who performed the tasks, and users who use the knowledge assets.

Organization Model: OM-4		**Knowledge Assets**				
Knowledge Asset	**Possessed By**	**Used In Process**	**Right Form?**	**Right Place?**	**Right Time?**	**Right Quality?**
Knowledge name (from OM-3)	*Agent (from OM-3)*	*Task (from OM-3)*	*Boolean (yes/no)*	*Boolean (yes/no)*	*Boolean (yes/no)*	*Boolean (yes/no)*
…	…	…	…	…	…	…

Table III.5: The outline of Organization Model worksheet 4 (OM-4)

The OM-5 worksheet is a feasibility and decision supporting document. The worksheet summarizes information derived from OM-1 to OM-4. It focuses on feasibility in terms of business, technique, and project. Then, the proposed actions are specified from the managerial commitment and decision making. This worksheet is important to give consensus to the knowledge project between the knowledge engineer, managers, experts and knowledge users.

Organizational model (OM-5)	Checklist for Feasibility Decision Document
Business Feasibility	*Problem and opportunity in business view (e.g. added value, change management, etc.)*
Technical Feasibility	*Problem and opportunity in technical view (e.g. technology, complexity, etc.)*
Project Feasibility	*Problem and opportunity in project view (e.g. commitment, resource, communication, etc.)*
Proposed Actions	*Focus, target, expected results, actions, risks*

Table III.6: The outline of Organization Model worksheet 5 (OM-5)

In order to understand the global view of Organization Model, figure III.8 illustrated the inter-connection diagrams of the worksheets. Each worksheet in this model is correlated to each other. The output from one worksheet was transferred to another worksheet.

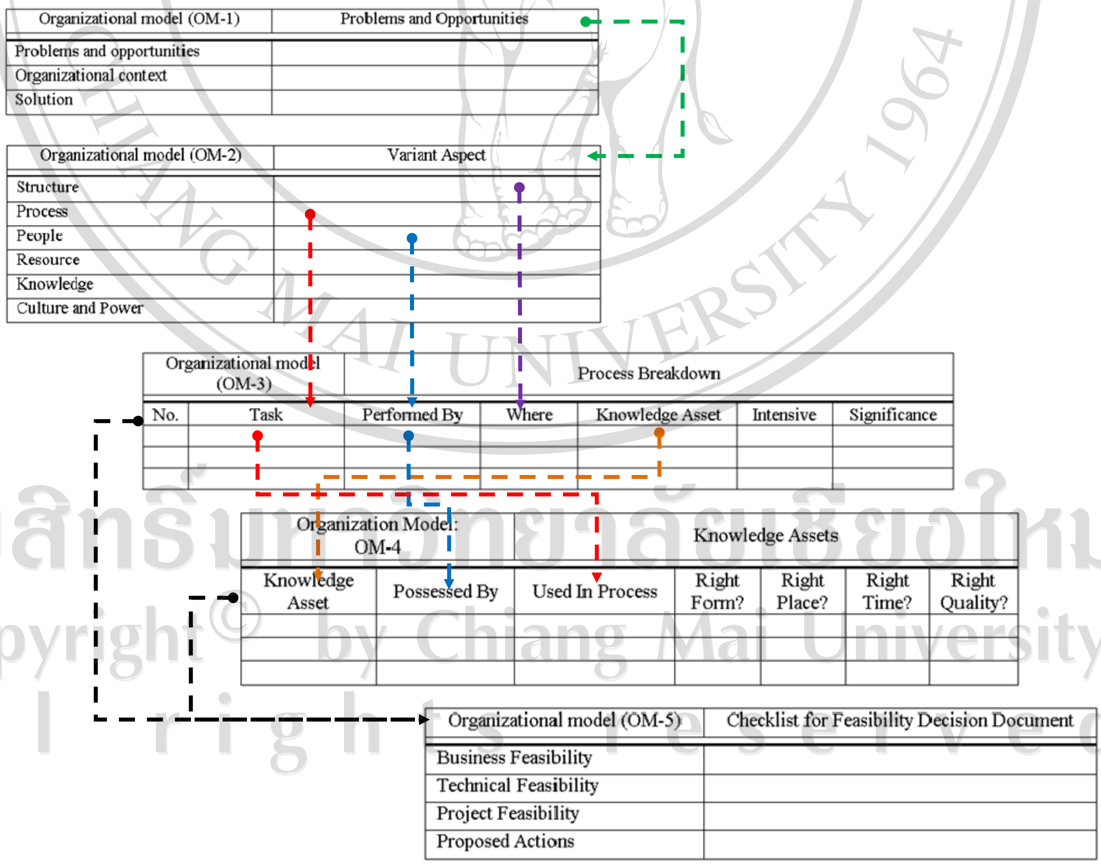

Figure III.8: Inter-connection diagram of organization model worksheets

The output that transferred from this model to the task model is a list of knowledge-intensive tasks which have consensus from the cluster members. The result of these worksheets will be presented in the next chapter.

III.3.1.3. Task Model

The task model is a refinement of knowledge intensive tasks identified in the organization model. For investigating a task, three viewpoints are concerned in this model. *The functional view* divides a task into subtasks: input and output. The s*tatic information structure view* is a description of the information content and structure of objects that are handled in the task. *The control view* (or dynamic view) provides understanding about triggering events, decision-making points, and other knowledge about the time aspect. The task model comprises two worksheets as shown in figure III.9.

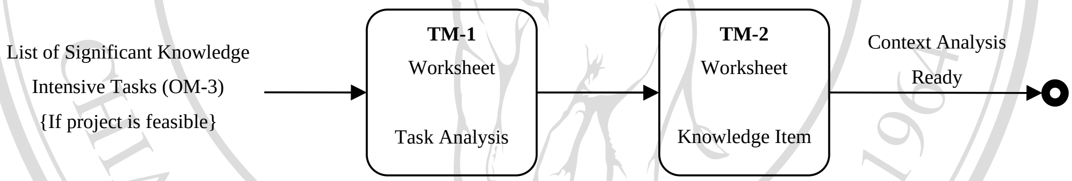

Figure III.9: Task models roadmap (TM-1 and TM-2)

The task analysis will be performed if the knowledge project is feasible (from OM-5). Each task from the list of selected knowledge intensive tasks would be analyzed in this model. The details of worksheet TM-1 is presented in table III.7.

Task Model (TM–1)	Task Analysis Worksheet
Task	*Task ID and name (from OM-3)*
Organization	*Cluster organization involved*
Goal and Value	*Objective and value of the task*
Dependency and Flow	*Input tasks, Output tasks*
Objects Handled	*Input objects, Output objects, internal objects*
Timing and Control	*Frequency, duration, control, constraints*
Agents	*Agents involved*
Knowledge and Competence	*List of knowledge involved (from OM-4)*
Resources	*Resource in the cluster involved (from OM-2)*
Quality and Performance	*List of quality and performance for measuring task*

Table III.7: The outline of Task Model worksheet 1 (TM-1)

In the complex or structured task, for example the production processes, the UML diagram is an appropriate tool for representing each view. For instance, functional view can be represented by an activity diagram, static information structure view can be represented by a class diagram, and control view can be represented by state diagram. All of the tasks analyzed in this worksheet will be arranged for reviewing in TM-2 in view of quality of the knowledge. The TM-2 worksheet is a specification of the knowledge employed for a task, and possible bottlenecks and areas for improvement. Actually, TM-2 is a refinement of OM-4. This worksheet is very important for the analysis, as it concerns the bottleneck and improvement of the knowledge in the organization. The TM-2 worksheet is shown in table III.8.

Task Model (TM-2)	**Knowledge Item Worksheet**	
ID:	*Knowledge ID (from OM-4)*	
Name:	*Knowledge name*	
Possessed by:	*Member of cluster*	
Used In:	*Task ID and name (from OM-3)*	
Domain:	*Domain of knowledge used*	
Nature of the knowledge		**To be Improved**
Formal, rigorous	*Boolean (yes/no)*	*Boolean (yes/no)*
Empirical, quantitative		
Heuristic, rule of thumb		
Highly specialized, domain-specific		
Experience-based		
Action-based		
Incomplete		
Uncertain, may be incorrect		
Quickly changing		
Hard to verify		
Tacit, hard to transfer		
Form of the knowledge		
Mind		
Paper		
Electronic		
Action skill		
Other		
Availability of the knowledge		
Limitation in time		
Limitation in space		
Limitation in access		
Limitation in quality		
Limitation in form		

Table III.8: The outline of Task Model worksheet 2 (TM-2)

From this TM-2 worksheet, a set of the knowledge is analyzed in terms of nature, form and availability of the knowledge in the industry cluster. The outcome of this model is a guide for the knowledge engineer in order to carry on the knowledge model in the next level.

III.3.2. Concept Level

The concept level is comparable with the requirement generation phase in the KMS development framework, which is used to obtain the required information for problem solving in the industry cluster. One common way for knowledge elicitation is to directly obtain information from the domain expert, called *the direct method*. It involves directly questioning a domain expert on how they do their job. However, for this to be achieved, the domain expert has to be reasonably articulate and willing to share information. Another way is the *indirect method* which is used in case the information from an expert cannot be easily expressed directly. The tools which support this method are: card sorting, 20 questions, document analysis, etc.

This level consists of two main processes: knowledge modeling and collaboration analysis. The knowledge modeling aims at extracting the knowledge from the experts and repositories. The knowledge model represents the knowledge shared between the participants in the cluster. In contrast, the collaboration analysis aims at understanding the approach for sharing the knowledge in the cluster. The collaboration model specifies the type of knowledge, method, and willingness to share the knowledge in the organization. The details of these models will be discussed as follows:

III.3.2.1. Knowledge Model

The CommonKADS knowledge model consists of three types of knowledge: domain knowledge, inference knowledge, and task knowledge, each of which captures a related group of knowledge structures (called knowledge category). *Domain knowledge* specifies the domain-specific knowledge and information types mentioned in the KMS. An example of this type of knowledge is the action for solving a specific problem. *Inference knowledge* describes how to make use of domain knowledge. It gives a primitive reasoning step for a knowledge model. For example, selecting ceramic products for international trade fair exhibitions of the cluster members, *"Match"* inference could be used for matching class, attributes and features of objects to meet the goal of classification. *Task knowledge* describes goals and strategies which were used for realizing goals. Task knowledge can be divided into several sub-tasks. This task knowledge is required by cluster members for achieving the knowledge intensive tasks in the industry cluster. Figure III.10 gives a brief overview of the three knowledge categories, as well as an example of knowledge elements.

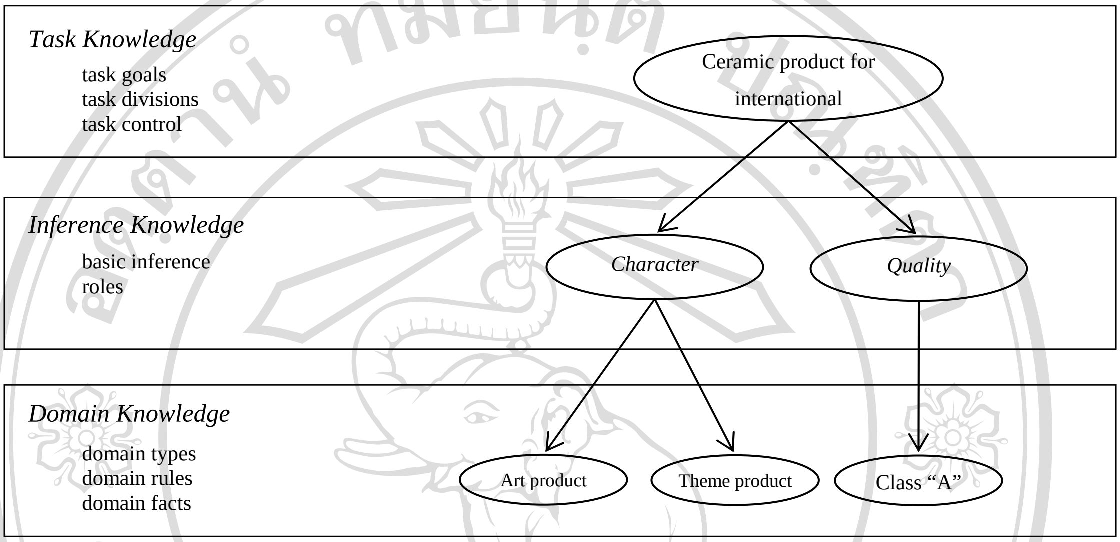

Figure III.10: Overview of knowledge categories and the knowledge model

The diagram in the figure above shows an example of the connection between each element of knowledge. The task in this knowledge is about selecting ceramic products for an international trade fair, while the domain knowledge is the attributes of ceramic products appropriate to the task. The inference provides the relationship between the task and the domain elements.

In order to extract three types of knowledge, CommonKADS provides a set of templates, which are the core of this knowledge engineering methodology, for eliciting the knowledge from the experts. Task templates form a common type of a reusable combination of model elements, which is a partial knowledge model [Schreiber 99]. It was designed for dealing with two groups of tasks in the system, analytical and synthetic tasks. *Analytic task* concerns the pre-existing artifacts in the system such as classification, diagnosis, monitoring task, etc. On the contrary, *synthetic task* concerns non-existing artifacts in the system such as design, planning, scheduling, etc. The overview of task types is presented in table III.9.

Task type	*Input*	*Output*	*Knowledge Type*
Analytic Task	***System observation***	***System characterization***	***System model***
Classification	Object features	Object class	Feature-class associations
Diagnosis	Symptoms / Complaints	Faults category	Model of system behavior
Assessment	Case description	Decision class	Criteria, norms
Monitoring	System data	Discrepancy class	Normal system behavior
Prediction	System data	System state	Model of system behavior
Synthetic Task	***Requirements***	***System structure***	***Elements, constraints, preferences***
Design	Requirements	Artifact description	Components, constraints, preferences
Configuration design	Requirements	Artifact description	Components, skeletal designs, constraints, preference
Assignment	Two object sets, requirements	Mapping set 1 to set 2	Constraints, preferences
Planning	Goals, requirements	Action plan	Actions, constraints, preferences
Scheduling	Job activities, resources, time slots, requirements	Schedule = activities allocated to time slot of resources	Constraints, preferences
Modeling	Requirements	Model	Model elements, template models, constraints, preference

Table III.9. Overview of analytic and synthetic task types

The process of the knowledge elicitation could be divided into 3 main tasks: interviewing, transcription, and modeling. Firstly, the knowledge engineer selects the knowledge template that suits the task type. Then, the interview complies with the objects in the knowledge template. The result from the interview is the transcript of the knowledge. In this stage, the knowledge engineer tries to repeat the interviewing process in order to totally complete the transcript. Finally, the transcript obtained will be represented in format of knowledge model. For instance, the classification template is one of the simplest templates for capturing knowledge in the analytic task. It is concerned with establishing the correct class for the object, which is based on characteristics of the object. This type of template is used for extracting knowledge model in the "Product selection for exporting" task, which is illustrated in figure 11.

The goal of this task is selecting ceramic products for export. The **Object** is the object for cauterizing which is a set of ceramic products. **Class** is the category of exhibition for ceramic products such as an international trade fair, road show, domestic market, local trade fair, etc. The **Attribute** is the characteristics of ceramic products that are usually defined in the cluster such as grading A, B, and C or ‘art product’, ‘theme product’, etc. **Feature** is an attribute-value pairing that applies to a certain object e.g. “international trade fair = ‘art product’ *and* ‘theme product’ which has factory grade = ‘A’ only”. The **truth value** is a categorized product that is matched to a required class.

The diagram III.11 gives an example of the knowledge modeling processes from the experts in the industry cluster. The processes comprise three parts: interviewing the expert by using the knowledge template, explicating the expert’s knowledge into the transcript, and modeling the knowledge model from the transcript. In the example, the knowledge about ceramic product selection for export was the focal point Product selection is classifying the ceramic objects’ features to match with the objective, which is an analytic task. Thus, the classification template was applied in the knowledge elicitation process. The outcome of this process is a set of structured knowledge that was represented in the format of knowledge transcripts. These transcripts are comparable with semi-final knowledge model of the task. The advantage of the transcript is that it facilitates the knowledge engineer in manipulating the knowledge. Afterward, the transcripts are converted into the knowledge models in task/inference/domain format. Finally, the modeled knowledge will be validated by the experts who are the owners of the knowledge. In the example, we represented the knowledge model with a Unified Modeling Language (UML) class diagram. However, there are many methods for modeling the knowledge such as topic maps, semantic networks, mind map, etc.

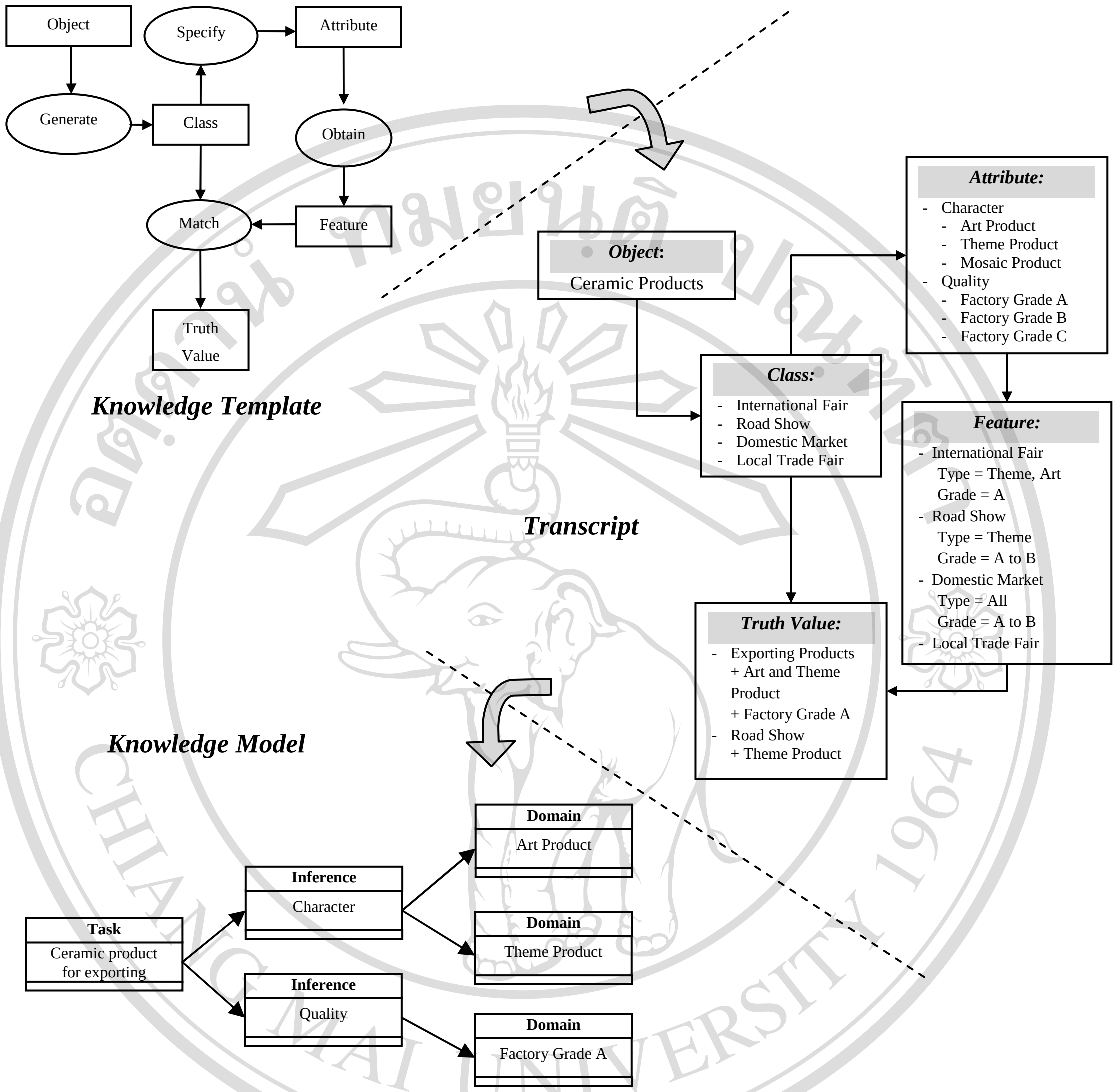

Figure III.11: The knowledge modeling processes

In the knowledge modeling/visualizing stage, UML was proposed as a standard notation for CommonKADS methodology. It comprises the activity diagram, state diagram, class diagram and use-case diagram. However, the methodology is not dependent only on UML. The "*topic map*", which is a standard for the representation and interchange of knowledge, can be used in the methodology. The topic map becomes the ISO standard and is formally known as ISO/IEC 13250:2003. The advantage of the topic map is that it is easier to read and understand by humans than the UML diagram. The inference class is transformed to the relationship between task node and domain node. Combining this with the topic map provides the semantic relationship to the knowledge model. Figure III.12 presents comparisons of knowledge representation by UML diagram and semantic topic map.

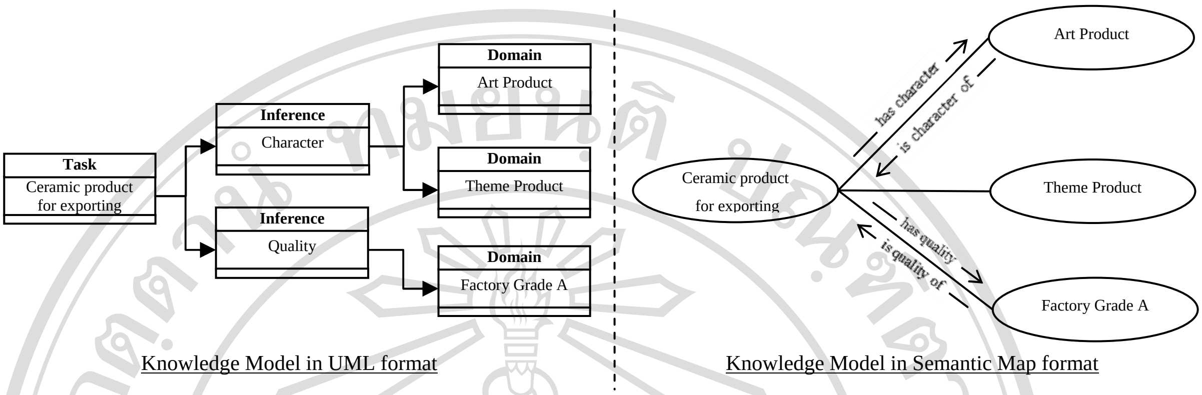

Figure III.12. Knowledge model in UML diagram and semantic map

Actually, topic maps are a form of semantic web technology (in the wider sense) and some work has been undertaken on interoperability between the W3C's RDF/OWL/SPARQL family of semantic web standards and the ISO's family of topic maps standards. Topic maps are also similar to concept maps and mind maps in many respects, though only the topic maps are standardized in this respect [Wang 07] [Natase 08]. The result from applying the knowledge model to the handicraft cluster will be presented in chapter 4 in UML class diagram format. Then, these knowledge maps will be stored in the knowledge-base of the KMS for further retrieval. The integration of the knowledge model in the knowledge management system will be illustrated in chapter 5.

III.3.2.2. Collaboration Model

The collaboration model specifies the information exchanged between the different members of the cluster. The collaboration model in this study is considerably different from CommonKADS communication model. The CommonKADS communication model focuses on the transaction of information exchange between agents working on the same task. However, due to the flat structure of the industry cluster and the loose relationship between the members, extracting the communication model of the cluster would be worthless. The communication in the cluster is varied by the characteristic of activity. In addition, in the “co-petition” relationship, cluster members share their knowledge depending on the level of the trust. Therefore, our collaboration model aims at analyzing the characteristic of collaboration in the cluster, and also modeling the knowledge sharing model of the industry cluster. It relies on two knowledge eliciting techniques, the in-depth interview, which is qualitative method, and the questionnaire which is quantitative method.

Interviewing consists of asking the domain expert questions about the domain of interest and how they perform their tasks. Interviews can be unstructured, semi-structured, or structured. Some interview methods are used to build a particular type of model of the task. The model is built by the knowledge engineer based on information obtained during the interview and then reviewed with the domain expert. In some cases, the models can be built interactively with the expert, especially if there are software tools available for model creation [Cooke 94].

The interviewing could be classified into three categories as follow. An *unstructured* interview is a general discussion of the domain, designed to provide a list of topics and concepts. A s*tructured* interview is concerned with a particular concept within the domain, a particular problem-solving skill or small group of skills. A s*emi-structured* interview is in between these two types. It is recommended to start interviewing with an unstructured interview for collaborative model identification. Then, the structured interviewing techniques, such as a problem-solving interview or dialogue, will be used for specifying collaboration model. In the final stage, the last interview attempts to validate the collaboration model of the organization.

In this study, the interviewing and questionnaire are considered as knowledge elicitation tools for modeling the collaboration of the industry cluster. It not only supports the collaboration modeling but also characterizes the collaboration, level of trust, etc. The objective of this model aims at answering the following questions:

1 What are the critical activities of the industry cluster? This question aims at analyzing the physical collaboration of the cluster members in various aspects.

2 What kinds of the information/knowledge are exchanged in the collaboration? This question concerns the type of knowledge shared in the collaboration, and also the willingness of knowledge sharers at different levels of trust in the cluster.

3 What are the internal and external factors that impact the cluster's collaboration? A set of questions are used to analyze the characteristics of collaboration in the industry cluster.

From the objectives of the model, the collaboration analysis could be divided into three main parts: activity (physical collaboration), information sharing and organizational characteristics. However, the organizational information is included in the questionnaire for analyzing the global view of the ceramic cluster. The diagram of the collaboration model is presented in the figure III.13.

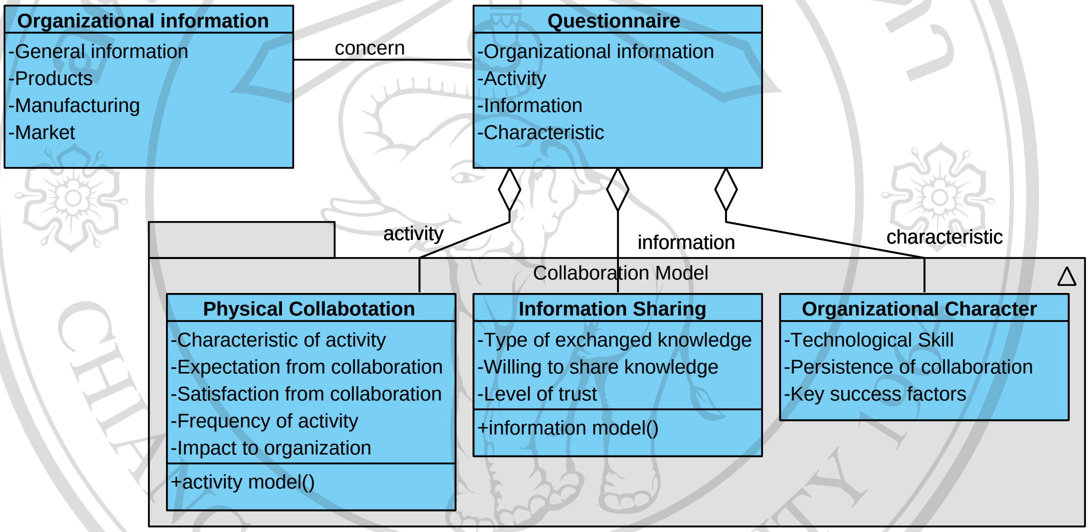

Figure III.13: The structure of collaboration model analysis

Accordingly, the outline of the questionnaire consists of four parts, as follows. A sample of questionnaire is showed in Annex C. The first part aims at acquiring the organizational information about the industry cluster. The second part focuses on analyzing the critical activities in the cluster collaboration. The third part aims at modeling the information and knowledge exchanging model of the members. Finally, the fourth part aims at analyzing the internal and external factors that affect the collaboration, such as information and communication technology, literacy or the environment of the collaboration. The detail of each part is described as follows:

<u>Part I: *Organizational information*</u> aims at acquiring general information such as characteristics of the organization, their products and markets. Although this part does not provide the concrete requirement collaboration modeling, it provides primary

statistical data about the industry cluster. The information extracted from this part was represented by using UML diagram as shown in the figure III.14.

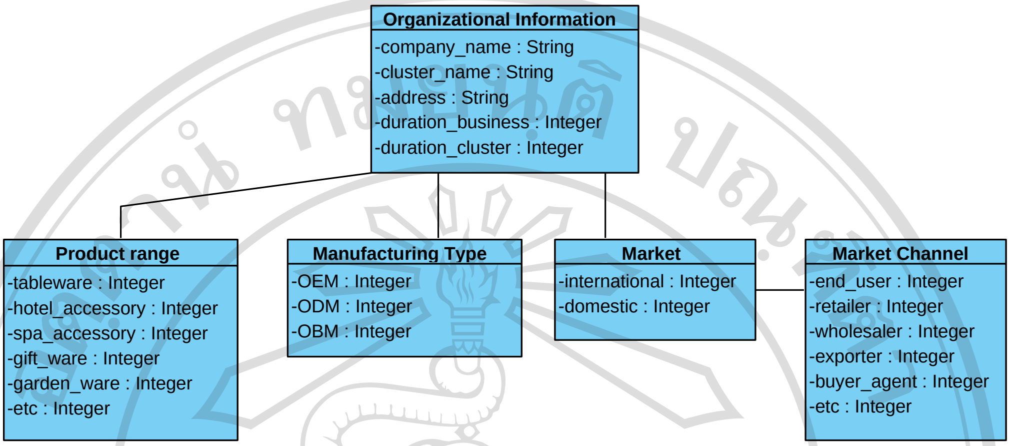

Figure III.14: The UML model of organizational information analysis

The diagram above presents required data mentioned in the questionnaire. The results from this part give a better understanding to the knowledge engineer about the members of the industry cluster. Therefore, this analysis was included in the questionnaire.

<u>Part II: *Physical collaboration*</u> aims at analyzing the main activities of the members of the cluster. The physical collaboration analysis concerns the different point of views on expectations, satisfaction, frequency, and impact. The list of activities can be obtained from the Organization Model 2 (OM-2) worksheet in the context level, or the interviewing of members. Then, the expectation and satisfaction from the collaboration will be compared with the intention of evaluating the value of the collaboration from the members' point of view. This value can be determined from the difference between the value of expectation and satisfaction. If the value of satisfaction is higher than expectation, this means that the collaborator gains benefit from the collaboration. On the other hand, if the expectation is higher than satisfaction, the members feel that this type of collaboration requires improvement. The total of differences also indicates the satisfaction of the members from participating in the industry cluster in the global view.

Another objective is analyzing the main activities performed in the cluster itself from the members' point of view. The evaluation concerns the frequency of collaboration and the expected impact of the activity on the industry cluster. The results from this analysis represent the value of the group of activities to the industry cluster. The details of activity analysis will be discussed in the next chapter, while the model of analysis is shown in figure III.15.

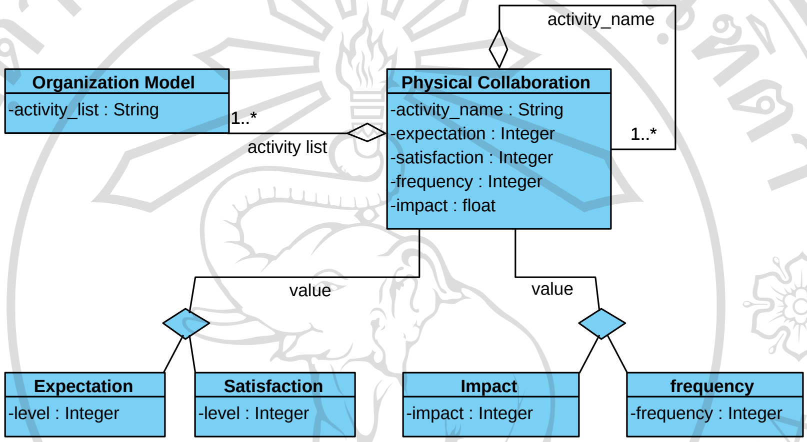

Figure III.15: The UML model of activity analysis

<u>Part III: *Willingness to share information/knowledge*</u> aims at analyzing the collaboration of the industry cluster from the information point of view. Thus, the objective of this part is modeling the structure of information and knowledge sharing in the cluster. The information structure modeling relies on the knowledge taxonomy which was explained in chapter 2. The questionnaire and interview intends to acquire shared information which matches with each type of knowledge. Moreover, the level and willingness of sharing are concerned for the modeling. The result of this part is the information model of the industry cluster which will be used for designing the collaboration services in the next level.

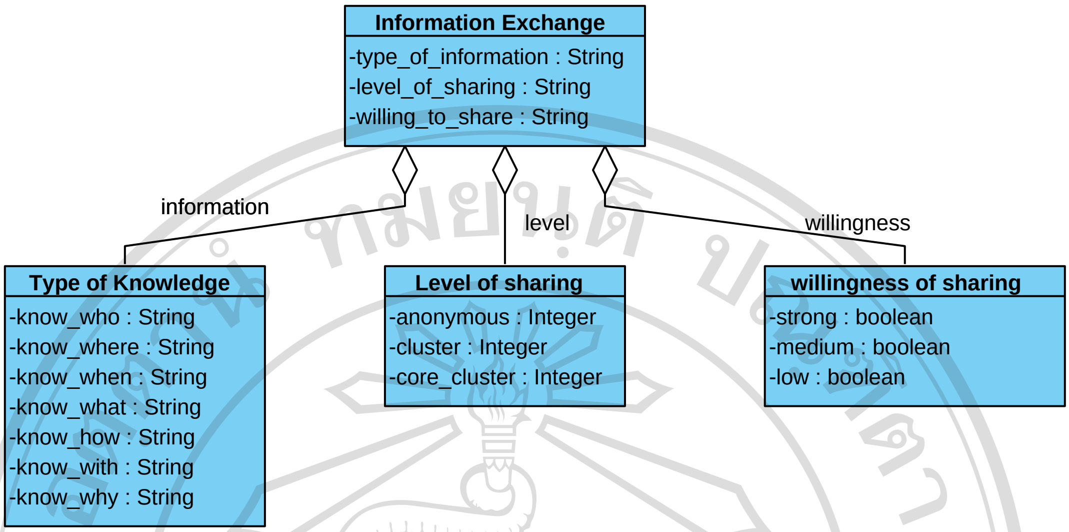

Figure III.16: The UML model of information sharing analysis

Part IV: *Characteristic of collaboration* aims at examining internal and external factors that affect the collaboration such as the technology aspect, problem solving techniques, strength of relationships, etc. These factors will be the criteria for designing the knowledge system. For example, the Information and Communication Technology (ICT) literacy of the cluster members will be used to specify the level of technology of the system. The conflict solving techniques and relationships are used for indicating the strength of the relationship. The key success factors of collaboration aim to analyze the present situation of collaboration by using 20 success factors of collaboration [Bruce 07]. The results give a better understanding about the collaboration in the particular industry cluster. Figure III.17 shows the criteria used in this part.

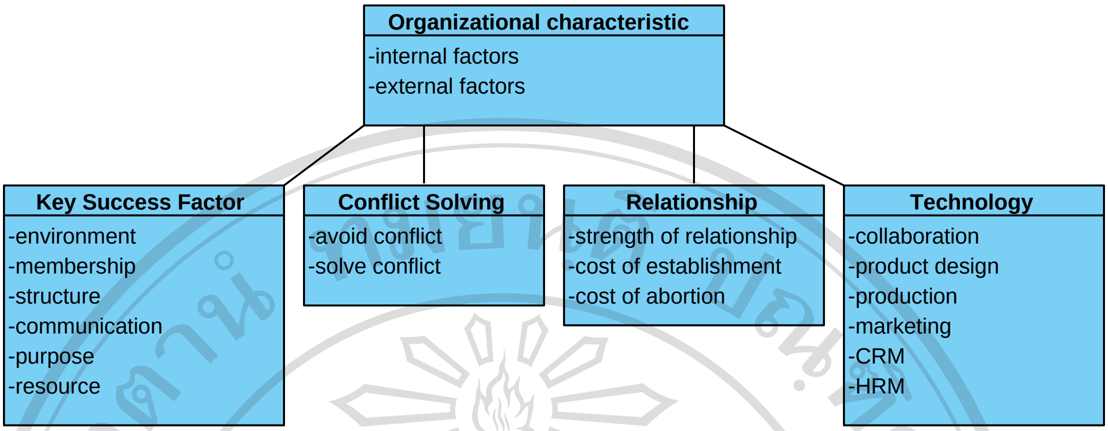

Figure III.17: The criteria for analyzing the characteristics of collaboration

In summary, the goal of this collaboration model is analyzing the critical activities of the members, the information sharing model, and the factors that affect the industry cluster. The models proposed in this study are an example of the collaboration analysis. However, the methods of analysis are not limited to these models. Thus, the interview is necessary for designing the questionnaire, while the questionnaire itself builds a consensus on the collaboration model.

III.3.3. Design Level

This level is comparable to the detailed design phase, which concerns the requirements, specification and architecture of the knowledge system. This level is necessary for transferring the responsibility from the knowledge engineer to the knowledge system developer. Although CommonKADS is also concerned with the design level in the methodology, it provides few explanations and is not concrete. Thus, this study emphasizes this level by adopting the software engineering theory in our methodology. In this level, we considered three views of the system development: specification view, scenario view and architecture view. Each view provides requirements and specifications from the knowledge engineering process to the knowledge system development process from the different aspects. The details of each view will be discussed as follows.

III.3.3.1. System Architecture

There are several models related to KMS architectures, as discussed in Chapter 2. Although there are differences in the function of each model, the fundamentals are similar. In this study, we adopted the concept of *"three-tier KMS architecture"* [Chua 04], which is the simplest architecture but concerns both knowledge management activities and collaboration. The model comprised three distinct services that are supported by knowledge management technologies. The architecture of the three-tiered model is shown in figure III.18. It is the assembling of three main services: knowledge services, collaboration services, and presentation services.

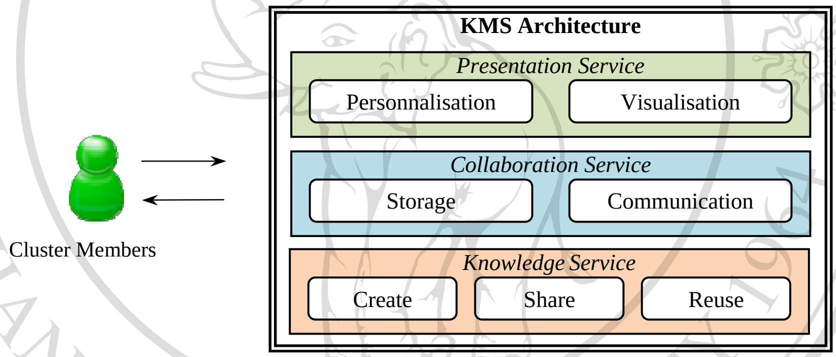

Figure III.18: The overview of three-tiered model

1 *Knowledge services*: intended to help cluster members achieve their goals of knowledge management. Three primary objectives are proposed to promote the process of generating new knowledge, encourage the flow of knowledge among cluster members, and ensure the ease of access to the knowledge repository [Martin 00].

- *Knowledge creation*: is the capability to capture and codify knowledge held by experts in the industry cluster. This process will be carried out by the domain experts or the knowledge engineer with knowledge elicitation techniques that are provided in CommonKADS.
- *Knowledge sharing*: is an important goal of KM technology that supports the knowledge sharing process which is collaborative tools, such as shared spaces, wiki, calendaring, workflow management service, etc.

- *Knowledge reuse*: is a synonym for "information retrieval" in the information management literature. The emerging technology aims at enhancing search capabilities as users require, and automatic generation of meta-data [Marwick 01]. An example of technology that supports the knowledge reuse process is semantic search.

2 *Collaboration services*: refer to the basic technology platform and features needed to implement KM. The purpose of these services is facilitating the collaboration among the cluster members. The two main infrastructures provided by technology are storage and communication.

- *Storage*: known as *knowledge repository* such as drawings, audio, video or multimedia documents. The knowledge server which allows users to build content, create references and establish links among documents is technology that supports KM processes, particularly knowledge creation and knowledge reuse. This meaning of storage also extends to the database which is required for managing the knowledge behind the system.
- *Communication*: supports the collaboration and information sharing activities within the cluster. These services are designed with regard to requirements from the output of collaboration model. The communication service enhances the quality of communication of the cluster member on the focused activities.

3 *Presentation services*: concerns the interface between the user and the information /knowledge source in the KMS. It aims at visualizing the required knowledge to suit the knowledge user's preference.

- *Personalization:* involves gathering user-information and delivering the appropriate content and service to meet the specific needs of a user [Bonett 01]. This service refers to the rule that determines how users and content are matched, based on their attributes and values.
- *Visualization*: helps users better understand the information and knowledge available by making subject-based browsing and navigation easier [Marwick 01]. This service seeks to represent the knowledge in the right format for the usage.

III.3.3.2. System Scenario

Unified Modeling Language (UML) is a useful tool in order to create the system model. For breaking down system architecture in to different views, “4+1 Model” [Kruchten 95] is a famous approach for modeling a complex system. The 4+1 model (figure III.19) depicts five views with UML: logical view, process view, physical view, development view, and use case view. Each view presents one aspect of the system and each has a particular kind of UML diagram associated with it.

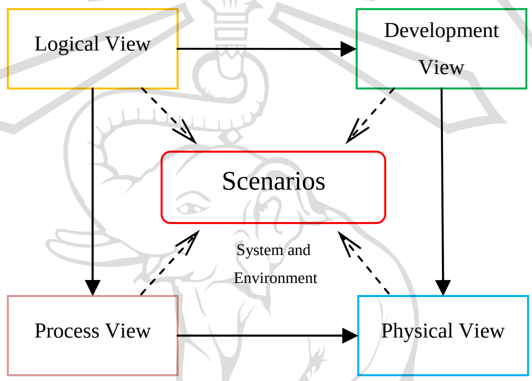

Figure III.19: The 4+1 model [Kruchten 95]

- The *logical view* shows the parts that comprise the system, as well as their interaction. The UML diagrams that show the logical view include: class diagram, state diagram, object diagram, sequence diagram and communication diagram.
- The *process view* describes a system’s processes. UML activity diagram represents the process view.
- The *physical view* models the system’s execution environment, which explains how to map software in to the hardware system. UML deployment diagram is used to model the physical view of a system.
- The *development view* describes the system’s modules, or components, including packages, sub-systems and class libraries. UML diagram that shows the development view includes: component diagram and package diagram.
- The use case view (*scenario*) shows the system functionality. It captures user goals and scenarios. This view is very helpful in defining and explaining the

structures and functionality in the other four views. The UML use case diagram provides the use case view.

The 4+1 view model allows various stakeholders to perceive what they need in the system architecture. The knowledge engineer can approach it first from the physical view, then the process view; cluster members, experts, and CDA can approach it from the logical view; and project managers and system developers can approach it from the development view [Kruchten 95].

III.3.3.3. System Specification

In this stage, Software Requirement Specification (SRS) which is IEEE standard (IEEE-830) has been used, in order to create a knowledge management system specification. SRS is a comprehensive description of the intended purpose and environment for software under development. It fully describes what the software will do and how it will be expected to perform. The core of SRS comprises five documents: requirement specification, functional specification, design specification, system specification, and test specification as shown in figure III.20.

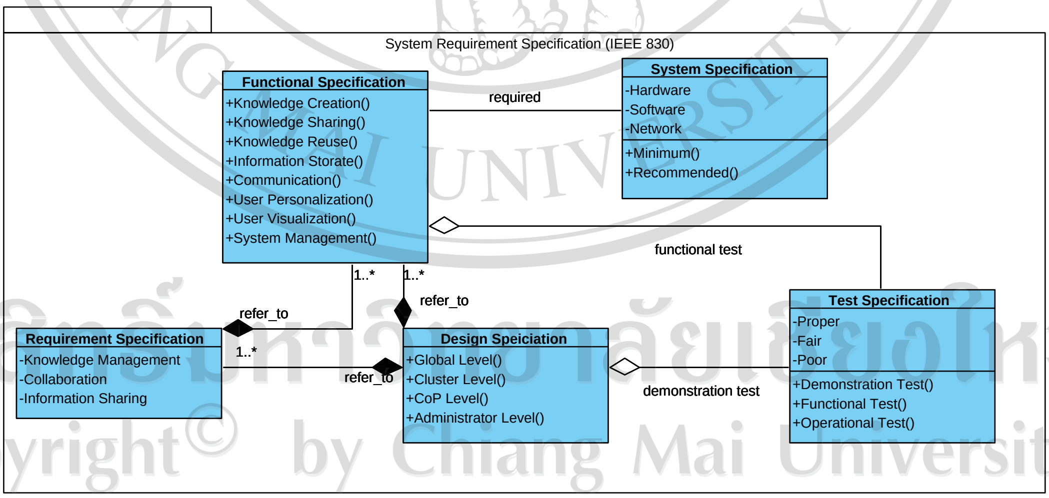

Figure III.20: Structure of Software Requirement Specification (SRS)

These documents are required for KMS development in the next level. They provide better understanding of the needs and constraints of the system to system developers and stakeholders. Requirement specification provides users' viewpoints for the system. Functional specification describes the requested behavior of the system. Design specification describes the organization of the functions/modules over the user interface. System specification describes hardware, software and environment that are required for operating the system. Test specification describes how to test each module, scenario and feed back for the system developer. These documents act like the medium for the knowledge engineer and system developer in order to develop the knowledge system in the next level.

III.3.4. Implementation Level

This phase is actually the KMS development process, also known as coding process. It aims at translating requirements and specifications into software product. Actually, the complexity of implementation phase varies on the defined specification in the previous section. The knowledge system implementation could be done by the simplest approach e.g. configuring existing system to suit with the user's requirement, until a more complex approach. The details of the implementation level will be described in chapter 5. After the prototype of knowledge system is created, the verification and validation of the system are required.

III.3.5. Verification and Validation

System verification and validation are essential for knowledge system development. A primary purpose is confirming that the developed system is matching with the specification and user requirements. Moreover, it aims at detecting software failures so that defects can be uncovered and corrected. The method for verifying the system is obtainable from the test specification in the design level. The test specification relies on three methods of system testing: Demonstration Test (DT), Functional Test (FT) and Operational Test (OT). The demonstration test focuses on testing the system by the role of user (e.g. anonymous user, support cluster, core cluster, CDA, and administrator). This kind of testing aims at verifying that the user requirements were supported by the developed system. The functional specification seeks to test the system function by function, which is indicated in the functional specification. This verification aims at detecting software failures in each function.

Lastly, operational test focuses on testing the system with the example test cases. It aims at verifying the system from the beginning to the end of operation in order to detect the errors between functions. Moreover, it also validates if the system complies with the industry cluster scenarios.

In general, test specification (which is functional testing) can be used for detecting the functional failure of the system. It provides the feedback about functional testing results to the system developer. Furthermore, the test scenario (which is non-functional testing) is also useful to check if the system is corresponding to the users' requirements. If provides the feedback in terms of suitability of the system for users' requirements to the system developer and knowledge engineer. These testing results are fed back to the requirement generation phase in order to modify the system specification and users' requirements. Finally, the revised system can be released for knowledge workers and experts in the industry cluster.

III.4. Methodology Implementation

The proposed methodology is the integration of knowledge engineering methodology, software engineering and cluster development technique. The combination of these three domains supports the knowledge engineer to develop the knowledge system for the industry cluster. The CommonKADS knowledge engineering methodology strongly supports the analyzing of organization, requirement and knowledge modeling, but less in knowledge system design and development. The software engineering methodology, such as the waterfall and spiral models, is widely used for managing and developing the software project. Cluster development tools such as Porter's diamond model, cluster mapping and collaboration analysis were used to overcome some limitations of the knowledge engineering methodology.

The core models of the methodology were separated into 4 levels, called model suites: context level, concept level, design level and implementation level. These 4 levels provide a step by step guide for the knowledge engineer from analyzing, modeling and designing until the development of the KMS. The model suite was illustrated in figure III.2.

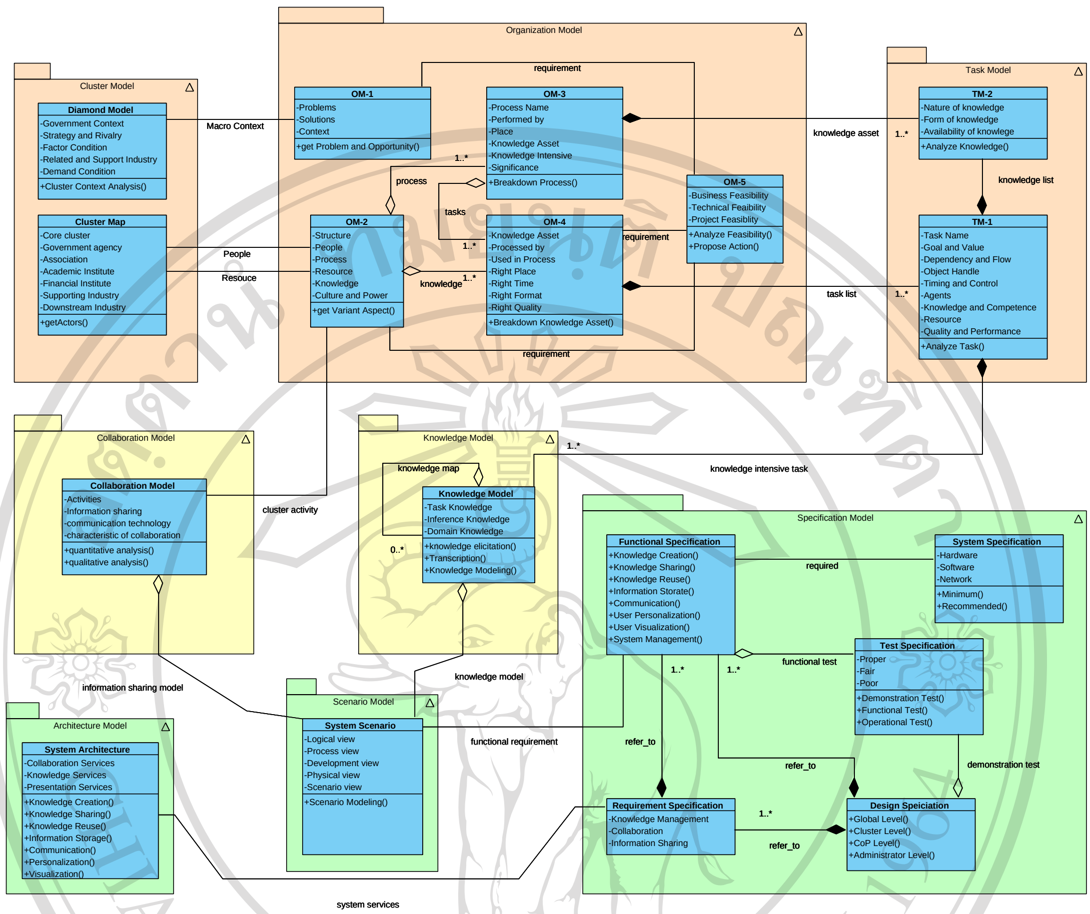

Figure III.21: The structure of the proposed methodology

The objective of the model suite is constructing the KMS for the industry cluster. Each level focuses on extracting information from the experts in different aspects. *Context level* aims to provide better understanding about the context of the cluster, knowledge intensive tasks and archetype of the industry cluster. As a result, this level provides the analyzed organization worksheets, task worksheets and the cluster map. *Concept level* aims at modeling the required knowledge, type of knowledge, pattern of sharing, and characteristic of collaboration in a particular cluster. The modeled knowledge and information sharing model are the result of this level. *Design level* aims to convert the results from previous models into requirements and specifications for the knowledge system. The output from this level is UML diagram, system architecture and specifications for system development process. *Implement level* is selecting information systems to meet the requirements and specifications that are defined in the design level. The final result of the model suite is the knowledge management system that complies with the organizational context, collaboration behavior, requirements and conditions of the industry cluster.

This model suite has been tested with “Cera Cluster”, which is the largest ceramic cluster in Thailand. The initial analysis was initiated at the core of the Cera Cluster level and the CDA. The results from the context and concept level will be described in the next chapter. Then, these results are considered as the criteria for generating requirements and specification for the knowledge system. Finally, the outcomes from design and implementation levels will be presented in chapter 5.