Chapter 3
Knowledge Management and Engineering

Knowledge Management is theoretically the business driven framework to improve effectiveness and focuses on knowledge workers. There are many theories and thinking concepts on Knowledge Management. These include for examples, Knowledge Creation focusing on innovation, Double Loop learning focusing on the defensive routines, the Fifth Discipline focusing on system dynamics, Intellectual Capital focusing on intangible value, and Learning in Action focusing on opportunity to learn. Regardless of the theory the main objective is to increase overall organization productivity through the use of knowledge.

Therefore, this relevant knowledge must be made explicit to others within the organization. This is where Knowledge Engineering plays an important role. . The methodology can prioritize, capture, analyze and represent knowledge in a suitable context where workers within the organization can utilize efficiently. In this research, Knowledge Management concept and Knowledge Engineering methodology are then applied to construct DMS design taxonomy and communication model to retain DMS design knowledge in long-term.

3.1 Knowledge

Data, Information and Knowledge are three often-encountered words that belong closely together. In table 3.1 data are the uninterpreted signals. Information is data equipped with meaning. Knowledge is the whole body of data and information that people use in action. Knowledge intensive company must have his/her knowledge management framework (knowledge identification, knowledge creation and acquisition, knowledge organization, knowledge codification and refinement, knowledge access, knowledge sharing and learning) to get his/her competitive advantage among competitors.

Knowledge [35] [36] is the whole body of data and information that people bring to practical use in action in order to carry out tasks and create new information.

Knowledge is a dynamic human process of justifying personal belief toward the truth and specialized because a knowledge worker develops his/her own expertise. Everyone has individual knowledge, skill and experience related to his/her action, problem solving and decision making opportunities. This type of knowledge, “Tacit Knowledge”, is the individual value system, critical component of collective human behavior and hard to articulate in formal language (table 3.2). An organization needs to explicit and develop common best practices among knowledge workers or communities of practice to support their core business.

Table 3.1: Distinction between data, information, and knowledge [35]

	Characteristics	**Example**
Data	Uninterpret Raw	…---…
Information	Meaning attach to data	SOS
Knowledge	1. Attach purpose and competence to information 2. Potential to generate action	• Emergency alert • Start rescue operation

Benefit of knowledge is faster decision making, increase productivity, and increase quality of decision making.

Table 3.2: Type of Knowledge

Tacit Knowledge (Subjective)	Explicit Knowledge (Objective)
Knowledge of experience (body)	Knowledge of rationality (mind)
Simultaneous knowledge (here and now)	Sequential knowledge (there and then)
Analog knowledge (practice)	Digital knowledge (theory)

Lundvall (and Tiwana) categorize how knowledge is used i.e. know-who, know-what, know-where, know-when, and know-with. There are two dimesions on understanding and connectedness of knowledge which shown in figure 3.1. the information is the understanding the relation among each data connected. The knowledge is the understanding of the pattarn among each information. And wisdom is the understanding the principle of the knowledge.

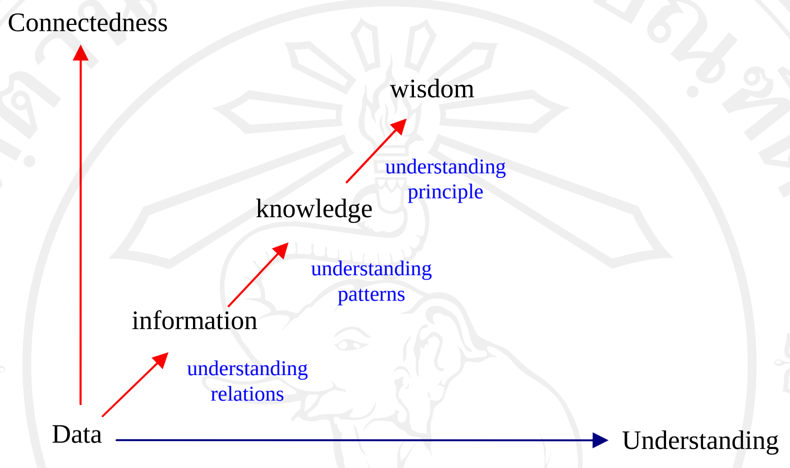

Figure 3.1: Knowledge Dimension

3.2 Organization Learning

Purpose of organizational learning is to convert individual knowledge into organization knowledge, to collaborate for exchanging intellectual material, and to collect intellectual material such as books, documents, etc. for asset [37]. There are five basic ingredients for a learning organization; understand how the company really works (system thinking), learn to be open with others (personal mastery), put aside their old ways of thinking (mental model), form a plan everyone can agree on (share vision) and work together to achieve that vision (team learning).

System thinking by story telling, storytelling is a good start for capturing knowledge and requirement in a different functional department. A good story can be modeled and told the story in this way allows others to be involved and to help develop a vision that is both individual and shared.

“Leader as teacher” is not about “teaching” people how to achieve their vision. It is about fostering learning, for everyone. Such leaders help people throughout the organization develop systemic understandings. Accepting this responsibility is the antidote to one of the most common downfalls of otherwise gifted teachers – losing their commitment to the truth.

Purpose of organizational learning is to convert individual knowledge into organization knowledge, to collaborate for exchanging intellectual material, and to collect intellectual material such as person, document, information etc. There are five basic ingredients for a learning organization [38]:-

- Team Learning : work together to achieve that vision
- Share Vision : form a plan everyone can agree on
- Mental Model : put aside their old ways of thinking
- Personal Mastery : learn to be open with others
- System Thinking : understand how the company really works

System thinking is looking for cause and effect in systematic way. Unlike reductionist, a good system thinker is someone who see events, system, patterns of behavior, and mental model operating simultaneously. For example, one can try to improve protection system on primary feeder by not only looking in great detail at individual relay setting for primary feeder but also the components of the feeder, type of feeder, location of feeder, type of customer, position of lateral on feeder, sectionalizer and fuse characteristic, fault indicator, position of lightning arrestor, minimum fault condition, overload load condition, operation procedure and considering the interactions between them.

System thinking technique such as ‘Storytelling’ is used for knowledge elicitation in this paper. It can help the organization collaborate between individual knowledge by reasonably story looking at the whole system picture. This method can generate the useful solution by aligning the main organization objective, reducing individual conflict and then compromizing the idea. When using ‘Storytelling’ technique to determine causes and effects for DMS lifecycle from distribution planning to decommissioning, the whole picture of distribution management system can be captured.

3.3 Community of Practice

A team of experts and knowledge workers in a specific domain is called "Community of Practice" [36]. In public service context, the communities of practice related for a specific task have been naturally built up within an organization. Sometimes community of practice can come across organizations. In a community of practice, team learning and problem solving are very necessary in helping each other at work. Other knowledge workers within the community of practice can share experience as well as exchange some ideas with the experts. These "Communities of Practice" are mostly informal and distinct from organizational units. They are a company's most versatile and dynamic knowledge resource and form the basis of an organization's ability to know and learn. Knowledge Management directly facilitates sharing and dissemination of domain knowledge as a key source of competitive advantage in the business world.

3.4 Knowledge Management and Methodology

Knowledge is justify, procedure, or belief about action which somehow commit with perspective and intension. Knowledge is "*actionable information*" [35]. *Actionable* refers to the notation of relevance and being available in the right place at the right time, in the right context, and in the right way so that users can bring it to bear on decisions, unlike information which simply gives us the facts. The border lines between data, information and knowledge are not sharp because they relate to the context of use. *Tacit knowledge* is very important and commonly referred to as unstructured knowledge (e.g. knowledge embedded in human mind through experience and jobs). However, one person's knowledge can be just another person's information. Therefore, it is necessary to have some kind of knowledge management.

Actually, there are mainly five theories of knowledge management in business framework which are:

- Double Loop Learning
- The Fifth Desicipline
- Knowledge Creation Company

- Intellictual Capital
- Learning in Action

<u>Double Loop Learning</u>

Rather than single loop learning emphasize on job description and specification that will confront a defensive routine in a mature organization. Agrilis decribes double loop learning which stress on explaination, reasonable discussion, ice-breaking techinique which is the second loop that organization need to cleary find it out.

<u>The Fifth Descipline</u>

There are five disciplines of the learning organization which are:

'*Personal mastery*' is a discipline of continually clarifying and deepening our personal vision, of focusing our energies, of developing patience, and of seeing reality objectively.

'*Mental models*' are deeply ingrained assumptions, generalizations, or even pictures of images that influence how we understand the world and how we take action.

'*Building shared vision*' a practice of unearthing shared pictures of the future that foster genuine commitment and enrollment rather than compliance.

'*Team learning*' starts with dialogue, the capacity of members of a team to suspend assumptions and enter into genuine thinking together.

'*Systems thinking'* is the fifth discipline that integrates the others and also needs the disciplines of building shared vision, mental models, team learning, and personal mastery to realize its potential. Building shared vision fosters a commitment to the long term. Mental models focus on the openness needed to unearth shortcomings in our present ways of seeing the world. Team learning develops the skills of groups of people to look for the larger picture beyond individual perspectives. And personal mastery fosters the personal motivation to continually learn how our actions affect our world.

Knowledge Creation Company

Nonaka and Takeuchi have put up a whole theory about knowledge management and its creation on the basis of the difference between '*tacit*' and '*explicit*' knowledge which can interchange in two dimensions (individual and organization) by Socialization, Externalization, Combination, and Internalization or (SECI) spriral model. Moreover, Nonaka has identified the physical and non-physical environment that support knowledge creation which called 'Ba'

Intellectual Capital

In the early and mid 1990s there was an increasing awareness in the business community that knowledge was an important organizational resource that needed to be nurtured, sustained, and if possible accounted for. Intellectual Capital was first defined as having two major components, information and knowledge capital, and structural capital. Information and knowledge capital was of course the organization's information and knowledge, but the informal and unstructured as well as the formal. The structural capital was of course the mechanisms in place to take advantage of the information and knowledge capital, the mechanisms to capture, store, retrieve, and communicate that information and knowledge.

Learning Organization

Garvin has introduced the learning process of people to acquire the knowledge which are three types of learning:

Intelligent Learning comes from '*Search*' (analysis and research on public sources or documents), '*Inquiry*' (framing and asking insightful questions through the interviews or surveys), and '*Observation*' (attentive looking and listening through direct contacts with users)

Experiential Learning or 'learning by doing' make people gain insightful knowledge through their experience. Certain types of knowledge come only from participation and involvement in doing things rather than studying and talking about them. Reflect and Review shall take place after the fact has occurred, such learning shall occur with lags. After being introduced to relevant concepts, theories and tools, people carefully select problems or simulation to test and apply new knowledge.

Along the way of processing, pauses need to be assured to evaluate progress, share learnings and make midcourse corrections.

Experimentation Learning which is '*Try-it-and-see*' (trial and error) by changing or modifying the condition of test or simulation, the results shall be observed and analysed and then draw a new conclusion. There are two kinds of this experimentation: '*Exploratory Experiment*' is to discover, to see what would happen if some things or some conditions pre-exist. And *'Hypothesis-testing Experiment'* is to prove, not discover; hypothesis is proposed and then proves it.

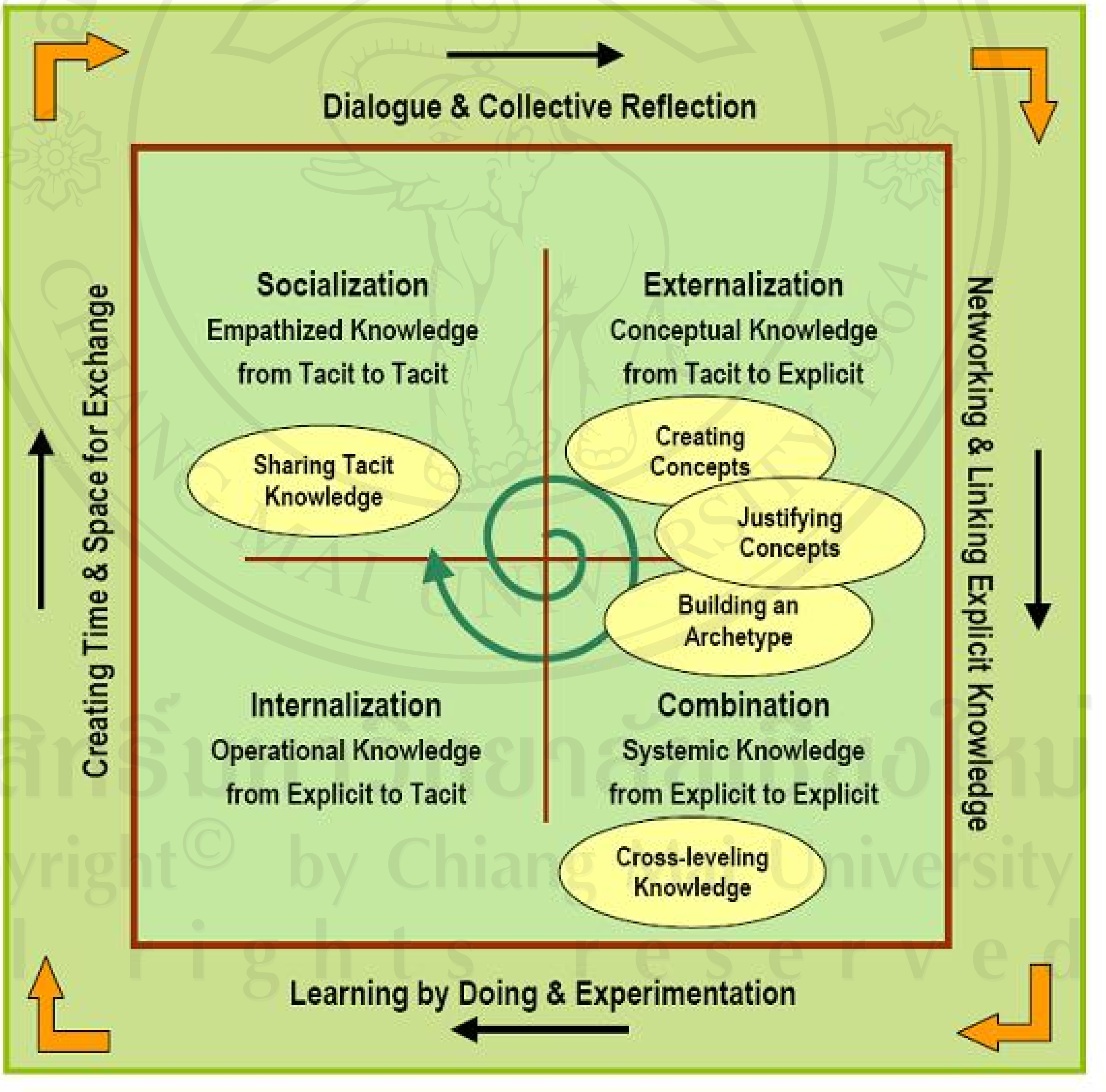

Figure 3.2: SECI Model [39]

This research is mainly based on knowledge creation, system thinking and learning organization theories. Nonaka present five stages for organization knowledge creating: sharing, creating concept, justify concept, build archetype, cross leveling. A knowledge project must be managed by learning from your experiences in a controlled *"spiral"* way. Nonaka has presented 'SECI' model (figure 3.2) for the five stages organization knowledge creating: sharing, creating concept, justify concept, build archetype, cross leveling. Defined knowledge as: justified true and skill, a dynamic human process of justifing personal belief and skill towards the truth, knowledge manager can lead and facilitate knowledge creation, accumulation and sharing by socially development a good Japanese word called *'Ba'* (physical and nonphysical dynamic environment where one shares a context with others to create meaning) for sharing context for knowledge creation and converting individual knowledge into organizational knowledge.

The knowledge management implementation should be introduced in figure 3.3. In principle, knowledge engineering is not some kind of mining from expert's head, but consists of constructing different aspect models of human knowledge. The knowledge-level principle: in knowledge modeling, first concentrate on conceptual structure of knowledge, and leave the programming detail for later. Knowledge has a stable internal structure that is analyzable by distinguishing specific knowledge types and roles.

3.5 Knowledge Engineering Methodologies

Knowledge engineering [35] [40] provides method and methodology to design and construct knowledge systems. The designed knowledge systems include expert system, decision support system, knowledge management system and knowledge archive. The knowledge engineering covers capturing, analyzing, validating and modeling a domain of knowledge. To implement the structured knowledge base, IT-based knowledge management system requires some effective design techniques and tools provided by some knowledge engineering methodologies. This section will review on four knowledge engineering methodologies:

- MOKA (Methodology and tools Oriented to Knowledge based engineering Application)
- SPEDE (Structure Process Elicitation Demonstrations Environment)
- AKEM (Application Knowledge Engineering Methodology)
- CommonKADS (Common Knowledge Acquisition and Design System)

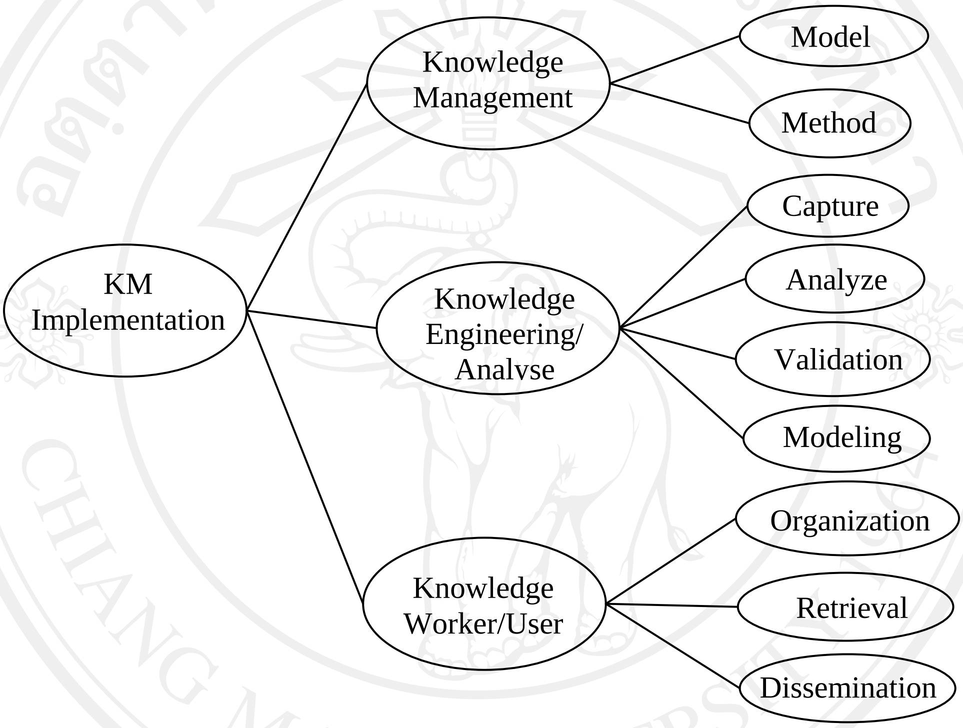

Figure 3.3: Knowledge Management Implementation

<u>MOKA</u>

MOKA support structuring and modeling knowledge about engineering design [41]. Moreover, it provides as well a standard way of storing knowledge that can reuse and maintenance of knowledge assets more feasible. The MOKA methodology emphasizes on two levels of knowledge representation: *informal model* and *formal model*. These models support the means of recording the structure behind the knowledge, not only the product and design process, but also the design justification as well. The informal model is assembled from five categories of knowledge types, described on forms known as ICARS forms (Illustrations, Constraints, Activities, Rules and Entities). These forms are linked together as shown in figure 3.4, and give a comprehensive and straightforward approach for placing structure upon the raw knowledge. This is prepared by introducing the knowledge to a right form, depending upon its type, and then identifying the link between the individual knowledge elements, the outputs of structure are informal model which can be stored and used for structuring the formal model.

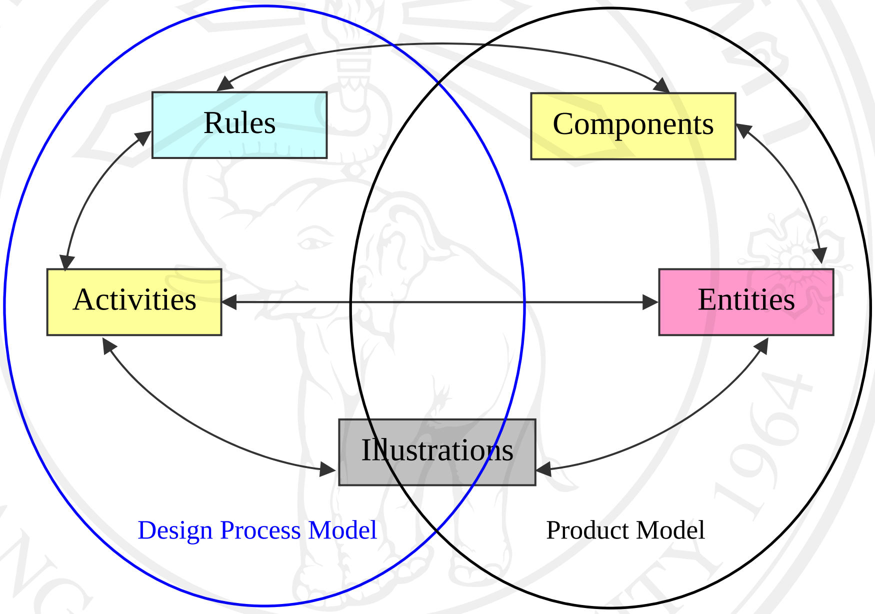

Figure 3.4: MOKA ICARE Forms

The formal model is relied on the formalized step. This step is consisted of two submodels i.e. *product model* and *process model* as illustrated in figure 3.5. the product model considers five views of the product: structure, function, behavior, technology, and representation views. The process model describes the activities and tasks of production. Activities contain description, input, constraint and method of production. Tasks are types of activities explained by a set of activities and their sequence.

This tool allows the user to build instances of the underlying meta-model. The knowledge stored in the informal model is transferred to the formal model by manual process, assisted by the MOKA tool until the contents of each ICARE form have been dealt with. This methodology is widely applied in the aerospace and automotive industries. It is one of the most well known methodologies for knowledge acquisition in the industry domain. The use of ICARE forms in the informal model, and the product and design process formal models, provided the user with templates specifically suited for the product design process, unlike more generic knowledge acquisition methodology [41]. We realized that MOKA is a solid methodology which mainly focuses on manufacturing context. However, the methodology did not provide much consideration to network and culture in organization.

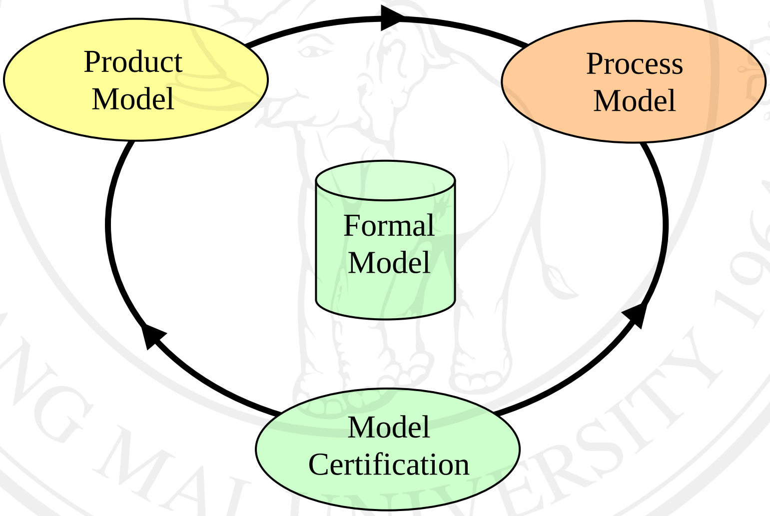

Figure 3.5: MOKA formal model

SPEDE

SPEDE is sponsored by both industry and academic. It presents a set of widely applicable methods and software tools to support and guide the business process engineer in the task of Business Process Improvement (BPI): mainly Business Process Reengineering (BPR), but including other improvement approaches. The SPEDE methodology is a combination of principles, techniques and tools taken form knowledge engineering and adapt to knowledge management. It provides an effective means to capture, validate and communicate vital knowledge to provide business benefit [42]

The concept of the methodology is designing mew business process in order to improve the organization knowledge by applying knowledge engineering techinique. This was done possible by dividing the activities into generally acknowledged high level BPR/BPI stages and providing a sequence at that level. In addition, it also provides core procedures for capturing the information created by the activities of each stage. The concept of this methodology is called 'Swim Lane' diagram as indicated in figure 3.6.

These processes help knowledge engineer to understand the context of the organization, such as business processes, people, resources, etc. Each step of the swim lane diagram is helped by the 'swim lane flowcharts'. They identify the project team activities for a BPI project sponsor, project manager, project team, and process modeling/simulation specialist [42].

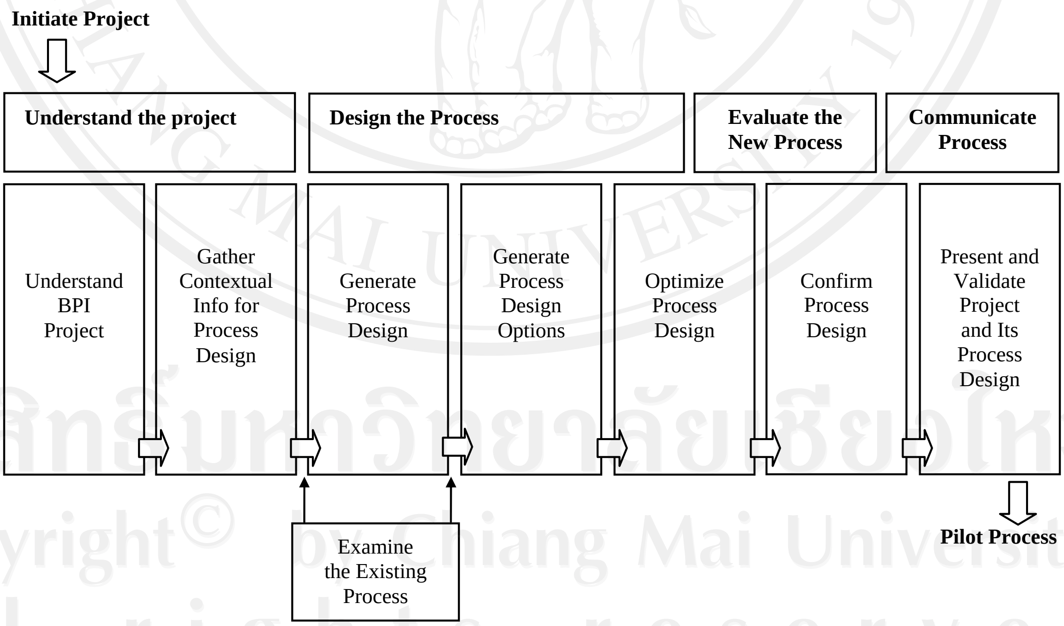

Figure 3.6: SPEDE swim lane diagram

For the knowledge acquisition process, SPEDE provides the Knowledge Requirement Templates (KRTs), which is a part of General Process Ontology (GPO), to facilitate and assist the knowledge elicitation process. KRT provides definition of the required knowledge to make role, agent, organization, and location associated with the 'activities' within a process. The GPO supports knowledge acquisition by enabling the definition of the structure of process knowledge earlier to acquisition. Table 3.3 shows an example of a KRT.

In summary, SPEDE is structured methodology which mainly focused on improving business processes within the company. The knowledge engineering methodology was adapted to enable the integration of process modeling and information modeling to facilitate rapid deployment of reengineered business process. SPEDE focuses on the context of the organization only on the process aspect. The network and culture of members in the organization is omitted in this methodology. In term of knowledge processes, this methodology more focuses on knowledge storage and retrieval in order to generate, simulate and analyze business improvement options, rather than assisting knowledge workers in the organization to achieve their tasks. This methodology is suggested for organizational KM project which focus on the business process of organization.

Table 3.3: An example of KRT of activities

Ontology	**Relationship**
Concept ID:	…
Name:	…
Attributes:	Start Time, Finish Time, Duration
Hierarchy Relations:	Has Sub-Activity, Has Parent Activity
Sequence Relations:	Ends Before Starts, Starts Bofore Ends, Starts before Starts, Starts after Starts, Ends after Ends, Meets, Contains
Resource Relations:	Uses, Produces, Consumes, Releases
Data Relations:	Has Data Input, Has Data Output
Other Relations:	Is Performed By,Has Location, Has Result

<u>AKEM</u>

AKEM is a collection of strategies and heuristics in knowledge capture, representation and application. A key principle of its development is the ease of practice and adaptation with emphasis on low ceremony and agility, considering the features of knowledge engineering and its dynamic context [43].

AKEM lifecycle model is summarized in figure 3.7. It organizes knowledge engineering projects through four phases: inception, elaboration, construction, and transition. Each phase is one or more iteration of eight activities with different degrees of emphasis and intensity: problem determination, scoping, analysis, development, and control. It recognizes the importance of knowledge management in knowledge engineering and stress the facility to trace back to the scope specification and conceptual concept in order to recapture or re-examine the previous modeling decisions. It also constructs traceability into deliverables in AKEM to enable links among stories, knowledge analysis, ontology, and deployment specification [43].

Problem determination: is the activity for classifying the character of problems to solve and examine the cost-effectiveness and feasibiltiy of a knowledge-based approach to solutions. The problem area is examined to identify and characterize the problem in the current application and system context.

- *Scoping*: the scoping activity in the domain perspective in AKEM delivers two main deliverables: knowledge resources (documents, interview protocols) and stories (knowledge use cases).
- *Analysis activity*: produces the knowledge constituent model and task hierarchy. The knowledge constituent model consists of the knowledge breakdown and the elaboration of each constituent. The knowledge breakdown seeks to modularize knowledge in a hierarchy structure, and the knowledge elaboration provides the description of each constituent in a program specification language to capture considered business reason.
- *Deployment*: is modeling the ontology underlying the story and knowledge constituent analysis. The knowledge is not adopted only for business reason and rules, but also the underlying meta-knowledge in the form of lexicons (context-term-role relations). The application specific constraints and rules are the special commitments to the lexons. The purpose is to maximize the reusability and versatility of knowledge resources, over different applications, time and versions. Ontology is extracted from knowledge resources, such as regulations, requirement specification, abstracted into term role tuples and

organized into an architecture reflecting the knowledge structure of the expert of the subject.

- *Extraction*: is a linguistic work in terms of its input and output. It works on natural language texts selected and generated in knowledge scoping and analysis: knowledge resources, stories and knowledge constituent analysis.

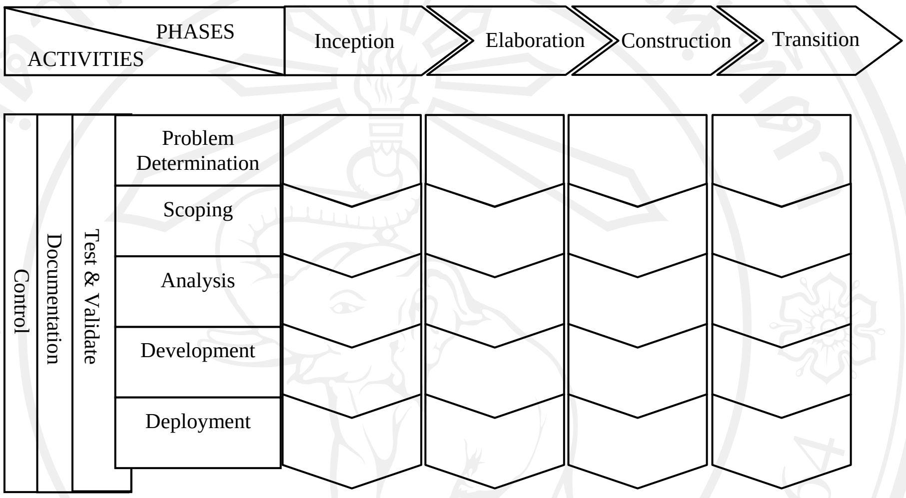

Figure 3.7: AKEM life cycle model

In summary, the AKEM methodology is mainly focusing on extracting knowledge from experts and userd in the organization for the purpose of developing a suitable knowledge system. The deliverables from each stage of the AKEM's framework are useful to the knowledge engineering to control, plan, and develop the knowledge system project. However, this framework is just a mirror of a software lifecycle development. The key characteristics of this methodology are application-oriented and ontology-based knowledge management system. Ontology modeling process, i.e. extraction, abstraction, and organization, were exploited for modeling organization knowledge for this methodology is still unclear. It did not provide the solid method than linguistic operation, key word hightlighting, and paraphrasing. In terms of the knowledge management activities, this methodology concerns all knowledge processes, but mainly focuses only on storage/retrieval and representation. Knowledge creation and transfer are briefly mentioned in the framework. AKEM methodology is suitable for a knowledge management project that relies on ontology and is metadata application-oriented.

CommonKADS

CommonKADS is a present version of KADS. This method has been developed since 1984 through two major CEC Esprit sponsored research projects. The methodology aims to support structured knowledge engineering. It indicated the opportunities and bottlenecks in the organizations, distributes and applies their knowledge resources, and so gives tools for corporate knowledge management. It also provides the methods to perform a detailed analysis of knowledge-intensive tasks and processes. CommonKADS supports the development of knowledge systems that support selected parts of the business process [35].

CommonKADS methodology offered a structured approach to break down and structure knowledge engineering process. It provided *CommonKADS model suite* for creating requirements specification for knowledge system as shown in figure 3.8. The method enabled a top-down approach and provided handles for quality control and feasibility assessment.

- ***Context level*** analyzes the organizational environment and the corresponding critical success factors for a knowledge system.
    - *Organizion model* supports the analysis of the major features of an organization, in order to discover problems and opportunities for the knowledge systemm establish their feasibility, and assess the impacts on the organization of intended knowledge actions.
    - *Task model* are the relevant subparts of a business process. The task model analyzes the global task layout, its inputs and outputs, preconditions and performance criteria, as well as needed resources and competence skills.
    - *Agent model* are the agents who are the executors of a task. It describes the characteristics of agents, in particular their conpetences, authority to act, and constraints in this respect. Furthermore, it lists the communication links between agents in carrying out the task.

- ***Concept level*** yields the conceptual description of problem solving function and data that were handed and delivered by knowledge system.
    - *Knowledge model* explicates in detail the type and structures of the knowledge used in performing a task. It provides an implementation-independent description of the role that different knowledge components play in problem-solving, in a way that is understandable by humans.
    - *Communication model* shows transaction between the agents involved in a conceptual and implementation-independent way, just as with the knowledge model.
- ***Artifact level*** combines the above levels together in the design model in order to construct the requirement specification for the knowledge system.
    - *Design model* gives the technical system specifications in term of architecture, implementation platforms, software modules, representational constructs, and computational mechanism needed to implement the functions used in the knowledge and communication models.

In terms of knowledge processes, CommonKADS methodology provides knowledge templates for supporting the knowledge modeling process, which constitute predefined reusable knowledge models and which have proven to work in the past, the details of knowledge templates will describe in the next topics. This methodology concerns all knowledge activities i.e. create, store/retrieve, share, and representation. However, knowledge sharing methods did get much attention in this methodology. The notions of network and organization culture are analyzed in organizational model in the context level. Although, this method did not provide a concreate process for handling a specific task, it is detailed enough to be able to apply in any knowledge intensive task. This methodology is recommended for KM projects that concern the knowledge exchange between agents in intra- and inter-organization.

Methodology Selection

None of the methodologies (Table 3.4) provide a tool to support activating process (KE methodology did not get much consideration in the initiating and maintaining application). However, most of the KE methodologies provide tools to support from identifying to packaging process, except MOKA which mainly focus on capturing and formalizing process by using informal and formal models. The main objective of MOKA is to improve manufacturing process.

Table 3.4: Comparing KE methodologies with the Coverage of Framework

Methodology	KE Project Management	Identify	Justify	Capture	Formalize	Package	Activate
MOKA	ICARE model	-	-	Informal Model	Formal Model	-	-
SPEDE	Swimlane Framework	Understand the Project	Understand the Project	Design the Process	Evaluate New Process	Communicate Process	-
AKEM	AKEM Lifecycle Model	Problem Determination	Scoping	Analysis	Deployment	Extraction	-
CommonKADS	CommonKADS Model Suite	Context Level	Context Level	Concept Level	Concept Level	Artifact Level	-

Each methodology provides KE tools (Table 3.5) to assist knowledge engineer in handle organization knowledge. AKEM is a general methodology which is adapt to any KM project. It relies on ontology modeling to represent knowledge, the knowledge elicitation process is still unclear.

Table 3.5: Comparing KE methodologies with provided tools

Methodology	KE Project Management	KE Engineerinf Tools	Knowledge Elicitation	Knowledge Representation
MOKA	ICARE Model	MOKA Modeling Language (MML)	Yes	Yes
SPEDE	Swimlane Framework	Knowledge Requirement Templates (KRTs)	Yes	Yes
AKEM	AKEM Lifecycle Model	Ontology Modeling	-	Yes
CommonKADS	CommonKADS Model Suite	KADS Templates	Yes	Yes

SPEDE, AKEM, CommonKADS, CommonKADS is focusing on structuring knowledge in the organization. From the comparison with three criteria, we agreed the

CommonKADS is the suitable methodology for applying in our reseach context because it emphasizes on structuring knowledge and provides a communication model which is able to fulfill the organization knowledge creation in the future.

3.6 KADS

KADS: Knowledge Analysis and Data Structuring [44] is a knowledge engineering methodology supporting the development of knowledge systems. In principle, a KADS knowledge model has three kinds of knowledge; task knowledge, inference knowledge and domain knowledge. Task knowledge contains knowledge about how elementary inference can be combined to achieve a certain goal. Task knowledge can commit to achieve a particular goal. Tasks represent fixed strategies for achieving problem-solving goals. The inference knowledge is to control knowledge that we abstract from the domain theory and describe the inference that we want to make the reason in this theory. The domain knowledge embodies the conceptualization of a domain for a particular application in the form of a domain theory. Domain knowledge can be viewed as a declarative theory of the domain. In fact, adding a simple deductive capability would enable a system in theory to solve all problems solvable by the theory.

CommonKADS is the EU de facto standard methodology for supporting design and implementation of knowledge systems. CommonKADS [44] or KADS (previous version of CommonKADS) has been broadly applied in power business, for instances, Knowledge Management for planning, operation, maintenance, pricing negotiation, asset management and regulatory issues. There are essentially 3 types of question which can be present in ‘Model Suite’ in figure 3.8 as follows:-

- Why? Why is a knowledge system a potential help or solution? For which problems? Which benefits, costs, and organiztional impacts does it have? Understanding the organizational context and environment is the most important issue here

- What? What is the nature and structure of the knowledge involved? What is the nature and structure of the orresponding communication? The conceptual description of the knowledge applied in a task is the main issue here
- How? How must the knowledge be implementing in a computer system? How do the software architecture and the computational mechanisms look? The technical aspects of the computer realization are the main focus here.

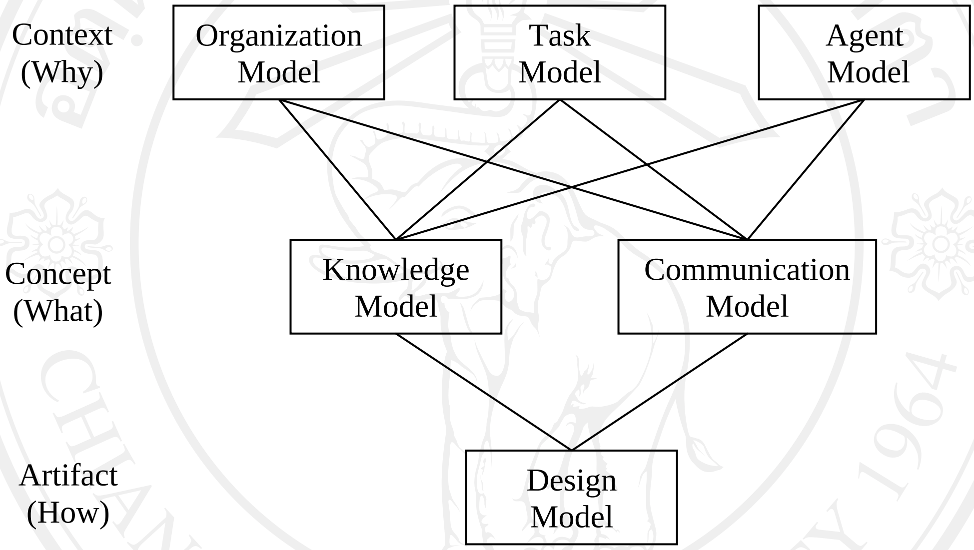

Figure 3.8: The CommonKADS model suite

This model suite can be classified to 3 levels and the useful templates for each level are provided for knowledge engineer to capture, analyze, validate, and model as follows:-

1. *Context Level*:- there are 3 mainly model in context level which are:-

1.1. *Agent Model*

Agent Model (AM) is a model of the 'kernel' team. Agents are executors of a task. An agent can be human, an information system, or any other entity capable of carrying out a task. The agent model worksheet (Table

3.3) describes the characteristics of agents, in particular their competences, authority to act, and constraints in this respect. Furthermore, it lists the communication link between agents in carrying out a task.

Table 3.6: Agent Model

Agent Model	**Agent Worksheet AM-1**
Name	
Organization	
Involved in	
Communicates with	
Knowledge	
Other competences	
Responsibilities and constraints	

1.2. <u>Organization Model</u>

Organization model (OM) supports the analysis of the major features of an organization, in order to discover problems and opportunities for knowledge systems, establish their feasibility, and assess the impacts on the organization of intended knowledge actions. There are five organization model worksheets which are:-

- OM-1 Problems and Opportunities Worksheet (Table 3.7)
- OM-2 Variant Aspect Worksheet (Table 3.8)
- OM-3 Process Break Down Worksheet (Table 3.9)
- OM-4 Knowledge Asset Worksheet (Table 3.10)
- OM-5 Feasibility Decision Worksheet (Table 3.11)

The first worksheet concerns about problems and opportunities, general context (mission, strategy, and environment) of the organization, and the potential solution on the problem.

Table 3.7: OM-1 Problem and Opportunities Worksheet

Organization Model	**Problems and Opportunities Worksheet OM-1**
Problems and Opportunities	
Organizational Context	
Solutions	

Table 3.8: OM-2 Variant Aspects Worksheet

Organization Model	**Variant Aspects Worksheet OM-2**
Structure	
Process	
People	
Resources	
Knowledge	
Culture & Power	

Then the second worksheet focuses on the structure, process, people, culture, resource, and knowledge within the organization. The process can layout in Unified Model Language (UML) activity diagram to visualize easier.

Third worksheet is the business process breakdown in specific task detail and identifies who, where, knowledge assets, intension of knowledge, and the indication of its significant.

Table 3.9: OM-3 Process Breakdown Worksheet

Organization Model		**Process Breakdown Worksheet OM-3**				
No.	Task	Performed by	Where?	Knowledge Asset	Intensive?	Significant

Table 3.10: OM-4 Knowledge Assets Worksheet

Organization Model		**Knowledge Assets Worksheet OM-4**				
Knowledge Asset	Possessed by	Used in	Right Form?	Right Place?	Right Time?	Right Quality?

The forth worksheet is a first-cut analyze of the knowledge assets which is very important for knowledge engineering and management to know that knowledge asset is in active use in the right form, place, time accessibility, and quality for the specific task or it needs an improvement.

And the last worksheet is to wrap up the key implication and presents the extensive checklists for producing the feasibility decision document.

Table 3.11: OM-5 Feasibility Decision Document Worksheet

Organization Model	**Checklist for Feasibility Decision Document Worksheet OM-5**
Business feasibility	
Technical feasibility	
Project feasibility	
Proposed actions	

1.3. <u>Task Model</u>

Tasks are the relevant subparts of a business process. The task model analyzes the global task layout, its inputs and outputs, preconditions and performance criteria, as well as needed resources and competences. There are 2 worksheets (table 3.12) used in task model which are:-

- TM-1 Task Analysis Worksheet
- TM-2 Knowledge Item Worksheet

Table 3.12: Task Model Worksheet

Task Model	**Task Analysis Worksheet TM-1**
Task	
Organization	
Goals and Value	
Dependency and Flow	
Objects Handled	
Timing and Control	
Agents	
Knowledge and Competence	
Resources	
Quality and Performance	

Task Model	**Knowledge Item Worksheet TM-2**	
Name Possessed by Used in Domain		
Nature of the knowledge		**Bottleneck/ to be improved?**
Formal, rigorous		
Empirical, quantitative		
Heuristic, rules 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 knowledge		
Limitations in time		
Limitations in space		
Limitations in access		
Limitations in quality		
Limitations in form		

2. *Concept Level*:- There are knowledge model and communication model in the concept level

2.1. Knowledge Model

Knowledge model contains a building block of task, inference, domain knowledge. It is a tool to clarify the structure of knowledge intensive task and it specifies the knowledge and reasoning requirement of the prospective system. We can divide knowledge model into 3 categories which are:-

Domain knowledge is the first category that specifies the specific knowledge and information type about the application. For example, the domain knowledge of application concerning DMS design would contains definition of the SCADA, DMS function, primary distribution system, and operation requirement as well as relationship between these types. The domain knowledge description is comparable to *'data model'* or *'object model'* in software engineering.

Inference knowledge describes the basic step that we use the domain knowledge. These are best seen as building block of reasoning machine. Two sample inference associate with DMS design could be '*propose*' the skeleton DMS design and *'revise'* the DMS design.

Task knowledge describes the goals of an application. 'How' these goals can be realized into subtasks or inferences.

Knowledge model can be partially reused in new application by using an available common task template. CommonKADS provides this common knowledge model template which distinguishes into two groups of task type:

'analysis' task and *'synthesis'* task. These are further subdivided into a number of task type based on the type of problem tackled by the task. In analysis task, system is preexists but not completely known. Analysis tasks take input data about the system and produce some characterization of the system as output. Contrary to synthesis task, the system does not exist. The purpose of the task is to construct a system description. The input of synthesis tasks typically consist of requirements that the constructed system should satisfy. Table 3.13 provides an overview of the main features of these knowledge model templates.

Table 3.13: Overview of Task Type

Task type	Input	Output	Knowledge
Analysis	*System observations*	*System characterization*	*System model*
Classification	Object features	Object class	Feature-class associations
Diagnosis	Symptoms/ complaints	Fault 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
Synthesis	*Requirements*	*System structure*	*Element, constraints, preferences*
Design	Requirements	Artifact description	Components, constraints, preferences
Configuration design	Requirements	Artifact description	Components, skeletal designs, constraints, preferences
Assignment	Two object sets,	Mapping set 1 → set 2	Constraints, preferences

	requirements		
Planning	Goals, requirements	Action plan	Actions, constraints, preferences
Scheduling	Job activities, resources, time slot, requirement	Schedule = activities allocated to time slots of resources	Constraints, preferences
Modeling	Requirements	Model	Model element, template models, constraints, preferences

2.2. Communication Model

Since several agents may be involved in a task, it is important to have a communication model for the specific task model in order to control and exchange practical information among agents.

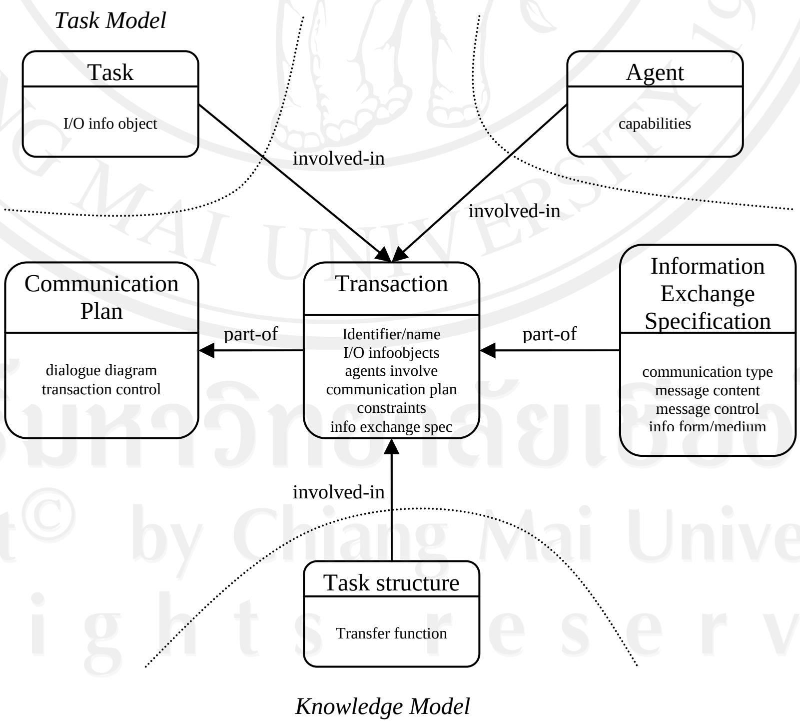

Figure 3.9: Overview of Communication Model

The overview of communication model is shown in figure 3.9 which can be divided into dialog diagram or communication plan (figure 3.10), transaction description worksheet (table 3.14), information exchange specification worksheet (table 3.15)

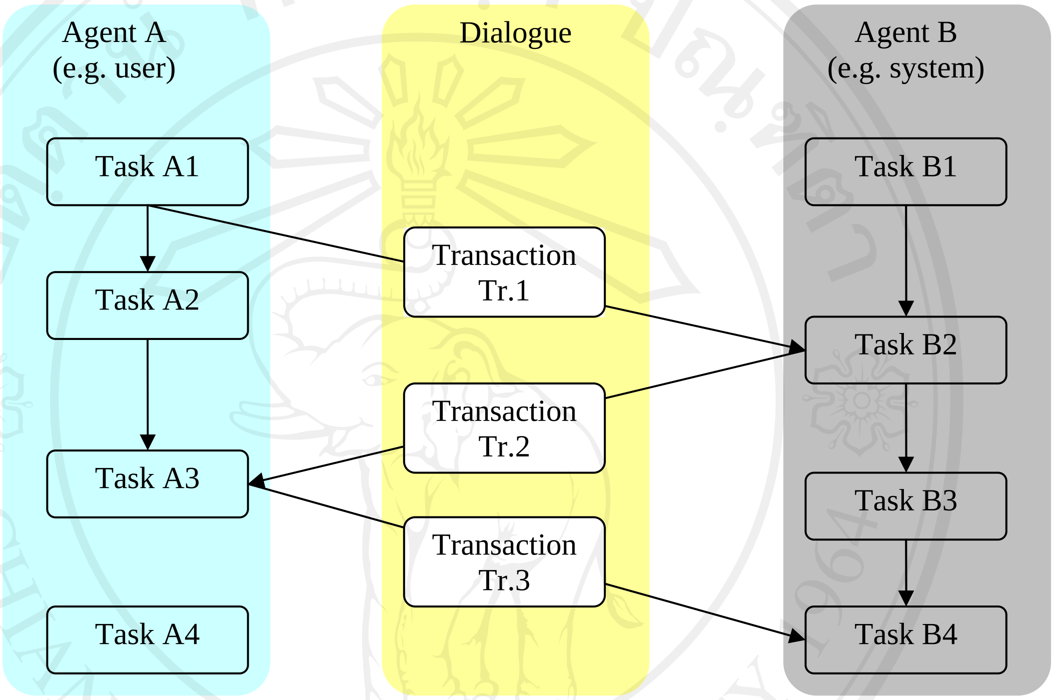

Figure 3.10: Communication Plan

Communication plan describes the full dialogue between two agents for the specified task. It aims to give an overview of all the need exchanges.

Table 3.14: Transaction Description Worksheet

Communication Model	**Transaction Description Worksheet CM-1**
Transaction	
Information Object	
Agent Involved	
Communication Plan	
Constraints	
Information Exchange Specification	

Transactions are the basics block for a dialogue which may consists of one or more messages. The heart of transaction is transmitting core information object to agent involved.

Information exchange specification gives role, form and medium information about message. In many cases it is helpful to not only transmit core information object but also to facilitate better understanding of the transaction by mean of additional explanation.

Table 3.15: Information Exchange Specification Worksheet

Communication Model	**Information Exchange Specification Worksheet CM-2**
Transaction	
Agents Involved	1. Sender: 2. Receiver:
Information Items	1. Role: 2. Form: 3. Medium:
Message Specifications	1. Communication Type: 2. Content: 3. Reference:
Control over Messages	

3. *Artifact*

The artifact is the design model of knowledge management system or knowledge system which gives the technical specification in term of architecture, platform, and software module. It should implement the functions that support in knowledge and communication model.

The short history of knowledge system:-

1965 GPS (General Purpose Search Engine)

1975 Rule-Base: MYCIN, XCON

1985 Structured Method: KADS

1995 Mature: CommonKADS

Knowledge engineering enable one to spot the opportunities and bottleneck in how organizations develop, distribute and apply their knowledge resources, and so gives tools for corporate knowledge management. Knowledge engineering provides the methods to obtain a thorough understanding of the structure and processes used by knowledge workers-even where much of their knowledge is tacit-leading to the better integration of information technology in support of knowledge work. Knowledge engineering helps, as a result, to build better knowledge systems; systems that are easier to use, have a well-structured architecture, and are simpler to maintain

There are six process roles in knowledge engineering and management which relate to knowledge provider, knowledge engineer, knowledge system developer, knowledge user, project manager, and knowledge manager. However, Knowledge Based System or Knowledge System in figure 3.6 has been used for the main components which are a reasoning engine and a knowledge base. There is some explicit representation of the knowledge included in the system. There are typically four major components of an expert system which are the dialog, inference engine, knowledge base, and explanation facility. The knowledge base contains the facts and rule of thumb in a given area. [45, 46]

Knowledge based system have 4 major components which are [47, 48]:-

- Dialog: user interface
- Inference engine: control structure of the system
- Knowledge base: contains facts and rules of thumb in a given area
- Explanation facilities: allow user to ask how and why.

Eventhough experts' tacit knowledge does not exist in explicit form; the most common way to deal with tacit knowledge is yellow pages which refer to the owner of the tacit part of knowledge. There are six process roles in knowledge engineering and management which relate to [35]:-

- Knowledge Provider : 'Expert' in application domain

- Knowledge User : makes use directly or indirectly of knowledge based system
- Knowledge Engineer : 'Analyst' elicits knowledge from experts and requirement from users
- Knowledge System Developer : is responsible for knowledge based system design and implementation
- Project Manager : is in charge of knowledge based system development project
- Knowledge Manager : formulates knowledge strategy

This is shown in Figure 3.11. From Propose and Revise template in previous subsection, Knowledge base that contains the facts and rules of thumb in DMS design can be developed and modified as the amount of DMS problem-specific knowledge existing within the organization.

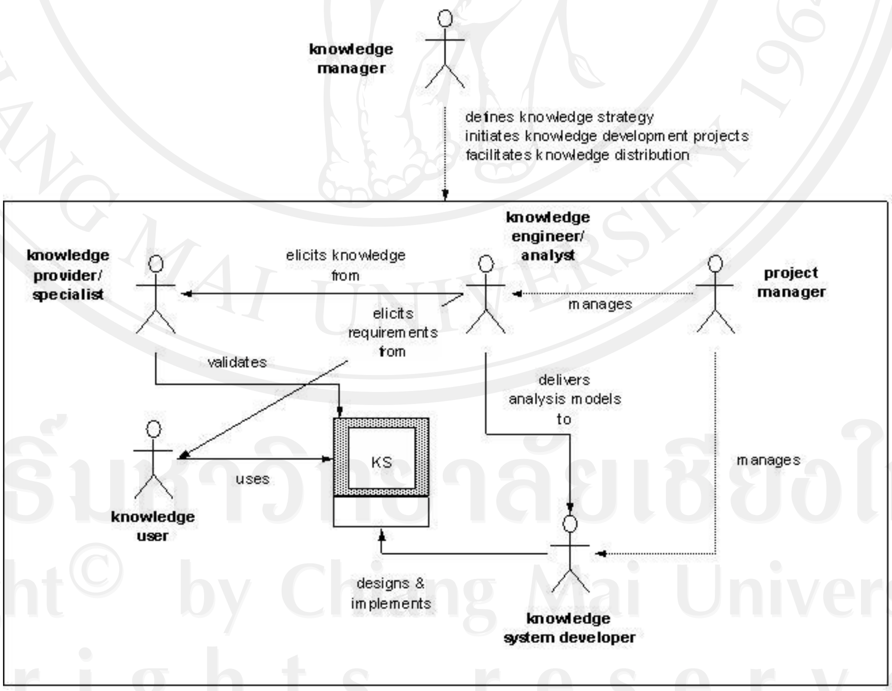

Figure 3.11: Knowledge Based System

3.7 UML Notation Used in ComonKADS

Unified Modeling Language (UML) is a set of standard notations for methodology developers which originated from an object-oriented viewpoint. It has received worldwide attention for the best view for software design industry. It provides a standard notation which could be imported to software development methodologies.

There are 4 Unified Modeling Language (UML) in CommonKADS.

1. *Activity Diagram*

This diagram control models the control flow and information flow of a procedure or process. It can model the organization process (worksheet OM-2) or model the control structure of a task method. Figure 3.12 shows the activity diagram for DMS design.

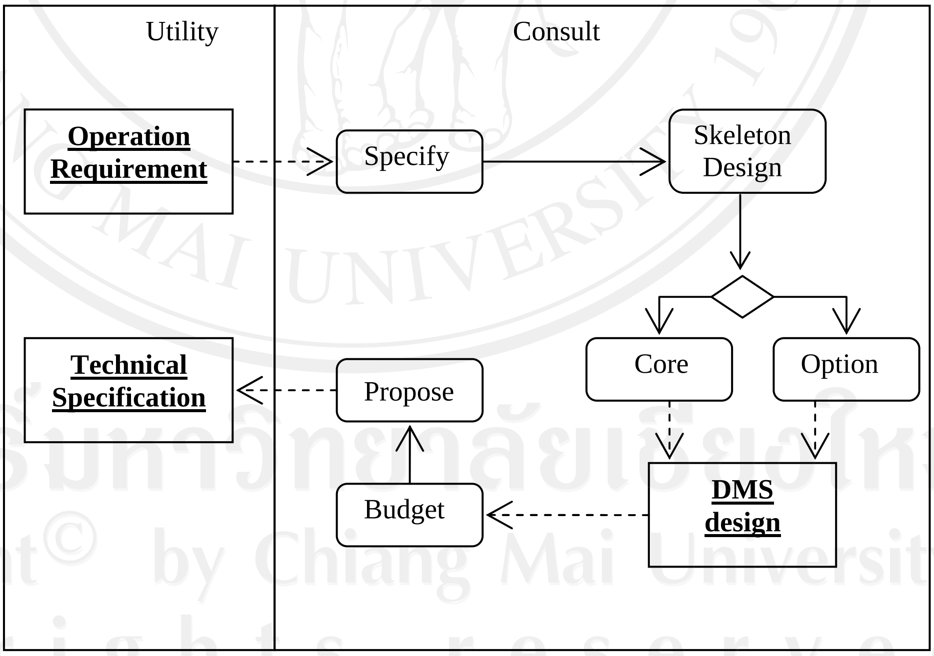

Figure 3.12: Activity Diagram

2. *State Diagram*

This diagram is a technique to model the dynamic behavior of a system that influence by the external events. *'State'* models the state of a system over a period of time. It is always a state of some object class. Not all objects have "interesting" states. During a state, *'activities'* and *'actions'* can be performed. *Activity* takes times and can be interrupted by events that cause state transition. *Action* is assumed to be instantaneous and cannot be interrupted. Within the state, 3 types of actions are defined:-

- *Entry*, carried out when a state is entered.
- *Exit*, done whenever the state is terminated.
- *Event-base*, only the execution of an action.

'State Transition' links between states. The syntax of this string is:-

<event> [<guard>] /<action> ^send-message(<class>)

The notation, <event>, causes a state transition to occur. The notation, [<guard>], is a condition on the transition. The notation, /<action>, is some process that always occurs when the state transition take place. And the notation, ^send-message[<class>], sends a message to some other object. Figure 3.13 shows the notation of state and state transition.

3. *Class Diagram*

Classes are used to describe the static information structure of the application domain. They represent groups of objects in the application domain that share a similar information structure. They are shown in boxes with 3 compartments: name, attribute, and operations. The relations between classes are association, generalization, aggregation, and composition. There are some differences between domain schema (in CommonKADS) and class diagram which are:-

- There is no relation in domain schema
- There is no rule type in class

- Classes in UML are called concept in domain schema
- Associations in UML are relations in domain schema

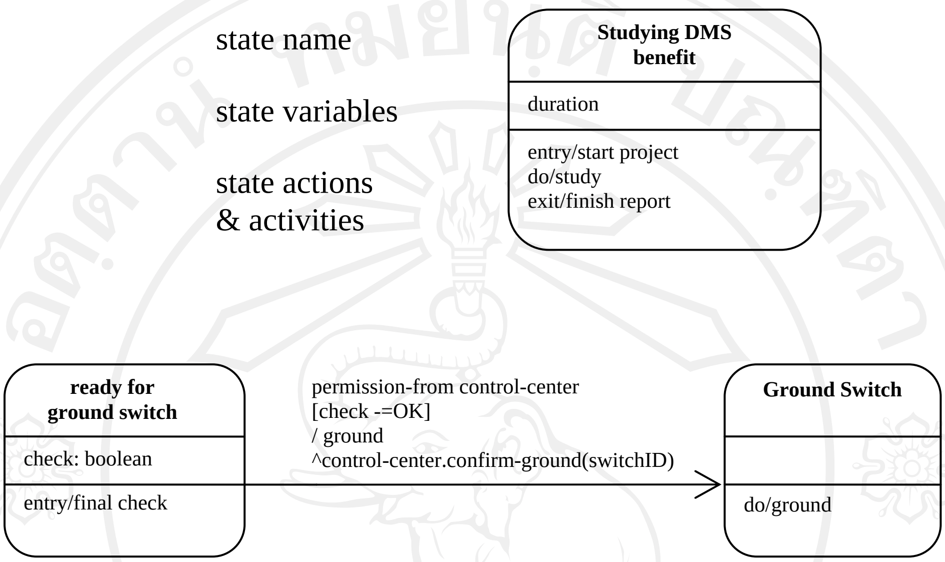

Figure 3.13: Notation for State and State Transition

Figure 3.14 shows the notation for distribution feeder class. It contains the attributes of the feeder which are type, size and so on.

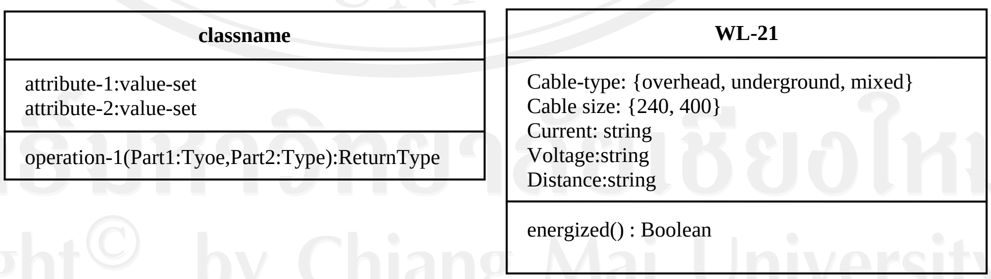

Figure 3.14: The notation for distribution feeder class

Association defines relationship between classes. The argument role can be specified in the link between object classes. The general notation for associations is a diamond symbol linked with object classes. The symbol can be omitted if it concerns a binary association.

Generalization is the common construct class diagram. Classes can be built in hierarchies with generalization. Class characteristics (attributes, operations, and associations) can be inheritance from superclasses to subclasses. The notation of generalization is an open triangular arrow.

Aggregation is that one class plays the role of 'aggregate' or 'whole' the other class. The notation is a line with diamond symbol. Figure 3.15 shows the example of DMS hardware in aggregation notation.

Composition is a strong form of aggregation. The 'part' cannot exist without the 'aggregate'. The notation is a black aggregation symbol.

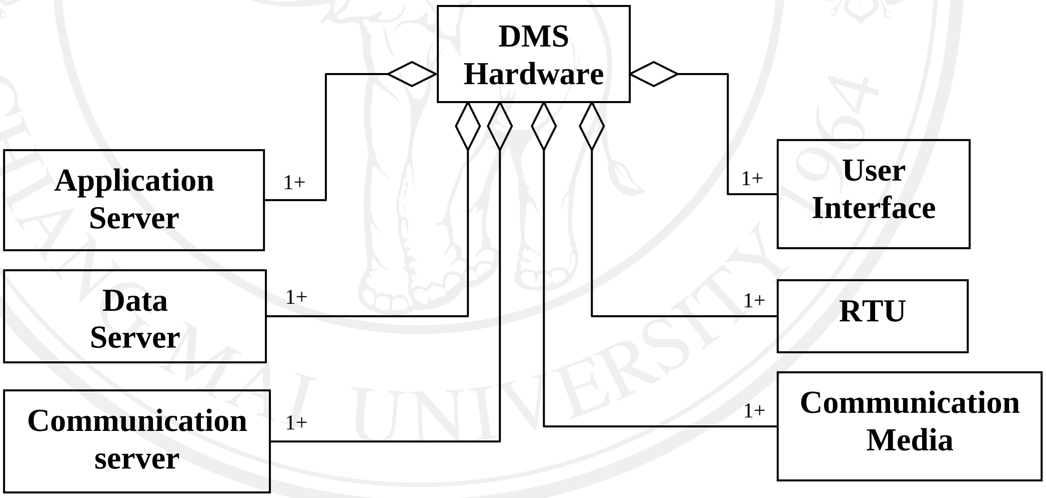

Figure 3.15: The notation of Aggregation

4. <u>*Use Case*</u>

Use case is useful as a communication vehicle between the developer and the customer and well fit with the agent model. It is a tool for the initial requirement engineering phase. There are two notations which are:-

Use-case

Use case is a service provided by a system. It is shown graphically as an ellipse with the name of the use case a label.

Actor

Actors are agent (humans or computer programs) that interact with the system. The most common relationship in use-case is the *interaction* relation between actor and a use-case. Figure 3.16 shows use case for technical assistance for DMS.

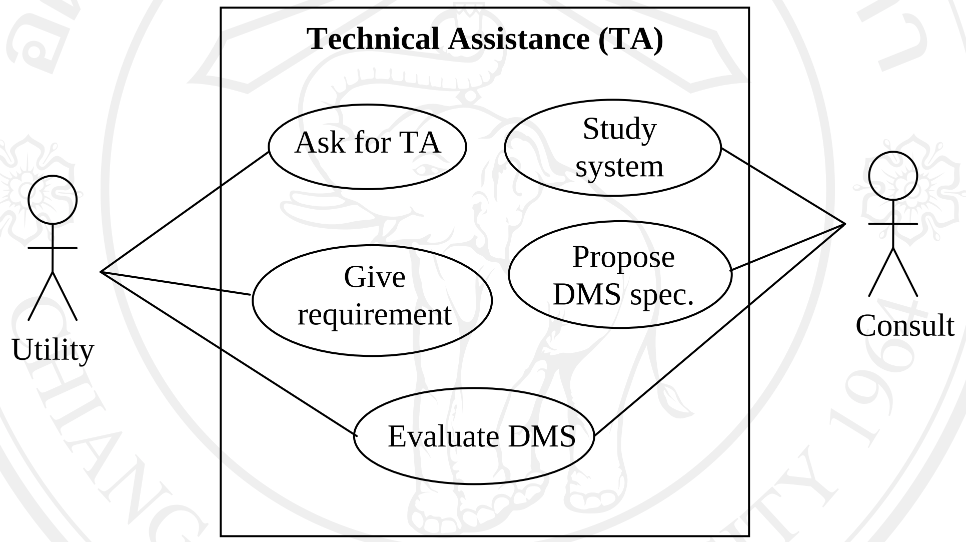

Figure 3.16: DMS Technical Assistance Use case