Development of a New Method for Generating Rational Requirement by Employing 2x2 Requirement Chart and OPM

Keiko Shimazu

Yokohama, Japan, 2013
Ph.D. Thesis
Graduate School of System Design and Management,
Keio University

Graduation Committee:

Chair: prof. dr. K. Takano (Keio University)

Members: prof. dr. S. Haruyamat (Keio University)

Emeritus prof*. prof**. dr. K. Furukawa (Keio University* / Kaetsu University**)

Emeritus prof***. Visiting prof****. dr. N. Aoyama (Tokyo University*** / Keio University****)

Abstract

The purpose of this thesis is to develop a new requirement identification methodology for constructing information systems having such characteristics that implemented functions are fully utilized, or in other words, only truly needed functions are implemented. This research is based on the observation that, in many real information systems, functional requirements written in such documents as RFP (Requirement For Proposal) were misleading. The reason why this happened can be thought as most information system designs were conducted with little attention to the ISO/IEC15288 standard for specifying the life cycle process of the system engineering or the ISO/IEC29148 standard for specifying requirements development. In more detail, the Stakeholder Requirement Definition Process and Requirement Analysis Process, which are often neglected in spite of their importance, produce as outputs ConOps (Concept of Operation) for the former and OpsCon (System Operational Concept) for the latter. The ConOps receives the Stakeholder Requirement statements which describe desired states brought by the information system to be developed based on use case analysis and context analysis. The OpsCon receives the Requirement Analysis results which describe functional requirements for realizing the ConOps. In particular, the use of the topmost figure of the context analysis results as the system overview figure of OpsCon is known as the key to identify a rational functional requirement. Accordingly, it is only possible to identify a rational functional requirement by preparing a highly qualified ConOps and by identifying OpsCon which is adequate for its realization.

On the other hand, in the real information systems development situations, the developers want to shorten the

upper stream design process and they tend to proceed to the identification stage of functional requirements with little use case analysis study.

In this research, we developed a new methodology to identify rational requirements with little extension of the total development period. In particular, we developed a representation method by using 2x2 requirement chart and OPM (Object Process Methodology). We applied the former chart in extracting requirements not for system realization but for the desirable state to be realized. Then we conduct both the use case analysis and the context analysis for the obtained requirements. We usually conduct the use case analysis is to identify the desired states to be realized and the context analysis to identify the undesirable states as well as the methods to avoid the expected drawback. In this research, we succeeded in developing a visualization tool to detect conflicts and/or negative use cases in the target information system by describing the use case analysis results and the context analysis results simultaneously using the OPM, which leads to identify the only truly needed functional requirements.

Acknowledgements

This thesis marks the end of a three-year period, during which I met many people that have contributed to my development as a researcher.

Especially, I want to thank the people of the SDM (Graduate School of System Design and management) at Keio University. I feel honored that you agreed to be members of the SDM.

To my students, some of you I spend more time with, drinking coffee or something stronger or even doing some work together. You inspired me to continue and spread my research.

Finally, I want to thank my best friend, Kayo-chan. She helps and supports me anytime.

Keiko Shimazu
Yokohama, 10 June 2013

Contents

Figurers and Tables

Chapter 1

Introduction

1.1 A Common Problem of Information System Products

Information system projects have been often evaluated based on their QCD (Quality, Cost, Deliver), since the investigation reported by the Standish Group in 1995 [1]. Excessive cost may especially cause projects to terminate depending on the amount. Generally, cost overrun against the budget occurs in unexpected tasks. Those are caused by the compensation for deficiencies discovered in the test-processes during the implementation period. It is almost impossible to accurately estimate the number and the degree of defects when designing systems. This causes the cost to exceed the budget.

Since the late 1980's, there's an on-going debate on Total Cost of Ownership (TCO), management and maintenance of information systems. At this period, outsourcing strategy for information system operation started to become popular [2]. This strategy made the information system operation cost stand out from the perspective of management.

Gruhl investigated the relationship between the costs in the Development Stage and the Concept Stage in 2

major aeronautics projects since 1978. According to this investigation, when the cost of the Concept Stage was under 5% of the development cost, all the projects exceeded the initial budget. In one case, the overrun cost itself was 170% of the budget (development cost was 270% of the budget). When the cost of the Concept Stage reached 10% of the development cost, the average overrun is around 20%. When this figure reaches 20%, there were not many cases where the cost exceeded the budget.

Forsberg et al. investigated the percentage of the costs from each period compared to the lifecycle cost in a number of large-scale complex systems. According to the results, 40% is used in the study period and development period, and the remaining 60% is mostly used in the operational period. Swanson et al. classified the costs in the operational period into support cost of user operation and system maintenance cost, and reported that the maintenance cost may sometimes equal or even exceed the entire development cost [3]. In the world of academia, there have been attempts to model the costs from previous projects and predict the future costs using these models. However, Glass et al. claimed that it is impossible to predict TCO [4]. They explain that since changes in social environment cannot be precisely predicted, these attempts aren't valid in the actual world.

There are also some papers that investigate the effect of operation cost. However, its effect on the entire Operation Period cost is not very large.

1.2 Agile Development

Agile Development is a collective term of methods for quickly and adaptively developing various information systems. In recent years, a number of agile development methods have been proposed. The number of cases where agile methods are actually being used in information development is also gradually increasing. Their common characteristic is that they all repeat actively and continually accepting changes and additions to the requirements of the system. This enables development of valuable system, which meets the true requirements.

The environment around information development is changing daily. Therefore, the focus of the important issue may change, or unexpected things may arise as a new requirement. Even though some elements may not be discussed in the Concept Period, engineers are increasingly faced with the issue of handling new requirements that have appeared during the development phase. This situation brings about a new concern that the operations in the Concept Phase may be wasted, and increases the desire to make the processes to satisfy the three elements mentioned above as simple as possible.

In the current industrial world, there are three factors that are always necessary for systems development process; (1)determine the system specifications early and realize them as a system in the shortest amount of time possible, (2)reduce the amount of waste in development, improve the quality and productivity, and maintain the adequate cost, and (3) assure that the clients receive investment effect and developers the expected compensation. It is logical to assume that agile development gained support due to these situations. There are, however, places where development operations are never converging, even the previous waterfall model lacks flexibility and is not very practical.

Prototyping is one of the major methods for agile development that is being widely used for service development. On the other hand, Yap reported that adopting this method causes cost overrun in many projects [5]. This work also introduces some lessons learned by identifying what kind of problems implementation processes face when requirements are incrementally identified. Talby investigated the process in which the problems found in the Development Stage are embedded and the relationship between the problems and the cost required to fix them. If the problems are created in the initial stages, it requires several to several hundred times the amount of cost to fix them. According to this work, the problems created in the Concept Stage require several times more cost than problems create in the Development Stage.

1.3 Systems Engineering Framework Standard

Therefore, Agile Development must be considered based on Vee model based on ISO/ISC12588, as well as non-information systems. INCOSE (International Conference of Systems Engineering) recognized that systems engineering should consider a system Lifecycle as all the processes, starting from the Study Period, moving on to the Implementation Period and then to the Operations Period, according to ISO/IEC15288. In the Study Period, an ambiguous image of the target system is converted into engineering specifications. Generally, we can minimize the defects found in the implementation period and in turn minimize the amount of work required to fix them by (A) identifying the requirements of the systems as specifications in engineering terms, (B) selecting a method that can achieve the identification in the Study Period and (C) accurately developing the system according to the selected method in the Implementation Period.

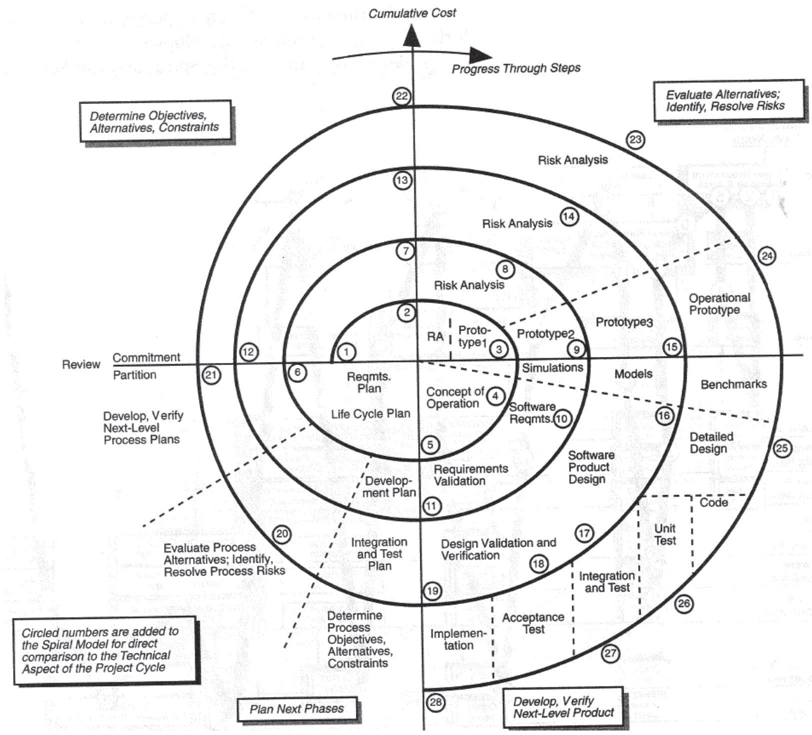
The defects found in the implementation period are mostly from a process (A) or (C). Defects from errors in process (C). In other words, they are from mis-identification of requirements or errors in the implementation method of (C).

The environment which surrounds the current information systems is changing all the time. Therefore the focus of the important problems may shift or new requirements which were previously unexpected may rise. This is causing a situation in which the developers have to answer the requirements that appeared during the implementation period. The developers are now worrying that the processes in the Study Period may be wasted, which in turn is forcing them to move on to the implementation processes without developing high-quality requirements in many information system establishment projects. The reason is that there is a trade-off between this and the desire to shorten the Study Period.

Nowadays, prototyping is one of the major methods of agile development that is being widely used for information development. On the other hand, Yap reported that adopting this method causes cost overruns in many projects [6]. This work also introduces some lessons learned by identifying what kind of problems implementation processes face when requirements are incrementally identified. Talby investigated the process in which the problems found in the implementation period are embedded and the relationship between the problems and the cost required to fix them [7]. If the problems are from the initial stages, it requires several to several hundred times the amount of cost to fix them. According to this work, the problems from the study period require several times more cost than the problems of the implementation period.

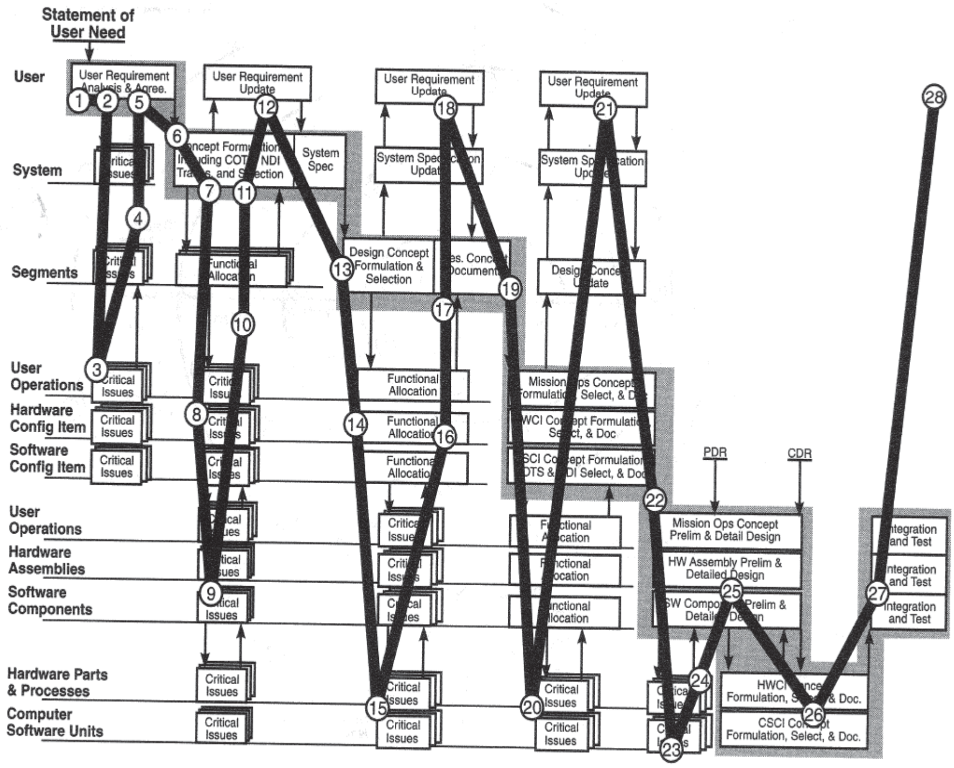
In systems engineering method, Vee model is a template for agile development with the system's robustness. Vee model is introduced after deciding from which phase to which phase in the systems Lifecycle will be processed using Systems Engineering methods.

On the left side of the model, we place the optimal phase from the viewpoint that decompose the component and interface group which would be necessary for the integrated system to be developed. On the right side, we place the optimal phase from the viewpoint which integrates the two. Vee model is used to decide on the optimal combination of activities to efficiently conduct the decomposition and integration. The goal of using the Vee model is to further ensure that regression of phase does not happen, which is the main objective of the system Lifecycle model. The waterfall model, which is well known in software development has a similar goal, but is more focused on completing the operations in each phase perfectly without any fault. This also means that the waterfall model aims to avoid fixing operations caused by the failures in the preceding phases. On the other hand, Vee model assumes that all the operations proposed in the spiral up would be conducted in every phase (Figure 1.1). In this phase, process activities such as an architecture design process, implementation process verification process are also conducted. In all the phases, all these operations are repeated. Depending on the stage of the phases, alternative operations such as simulation or prototype development may be conducted instead of the actual implementation operation. In other words, we expect that the processes for requirement determination for implementation in each phase are conducted using some method, and the results are reflected in the operations of the following phases (Figure 1.2). This is conducted efficiently using the template that determines the combination of activities that are conducted simultaneously across the processes for each phase, which is also known as the Vee model. In other words, Vee model can be described as a waterfall model from the front and as a spiral up from the side.



Modified [10]

Figure 1.1 Spiral up Model



Modified [10]

Figure 1.2 Spiral up Model Concept in Vee Model

1.4 Requirement Engineering Framework Standard

A number of cases are reported every year where ambiguous requirement definition is causing delays in construction and a cost increase in systems development [1, 2, 7].

There have been various requirement definition techniques developed. Most of them are focused on partial optimization with poor traceability and almost none of them are focused on the total system optimization.

Meanwhile, in the world of global standards, ISO/IEC29148（Systems and softwere engineering —Life cycle processes — Requirements engineering ISO/IEC29148） was established in 2011, aiming at improving requirement engineering quality thus avoiding construction delays and cost increase in the systems development. This standard complies with ISO/IEC15288, the Lifecycle Process Standard in the systems engineering to efficiently design large-scale and complicated systems. It is designed to achieve the total system optimization as well as the partial optimization at the same time. This applies to either occasion where the software component is regarded as a single system and where it is regarded as a part of a large-scale system. Being compliant with ISO/IEC15288, a fair amount of traceability could be secured once the system requirements are clearly defined.

According to the standards, the most important factor is not the methodology or technology being used, but the identification of the "as is" state of the domain and the "to be" state after the problems are solved. This clarifies what is being demanded. In other words, ConOps (Concept of Operations) focuses on the operation side of the processes of the services and organizing necessary information. Whether this has been thoroughly considered affects the essence of the requirements for the services, and determines the amount of deficiencies and the cost required to fix them. By properly preparing ConOps, we are able to select the optimal method to identify the system requirements.

Therefore in system development projects in fields such as aeronautics, which consist of large, complex systems that can provide safety, preparation of ConOps is the most significant process [8]. To process the contents more precisely, Roger cites the significance of elaboration of the requirement statements [9]. Tamai classifies the requirements analysis methodologies and cites the importance of an object-oriented approach [11].

Taking that into account, it's considered that defining the stakeholder requirements out of stakeholder needs and then defining the system requirements in a systematic and seamless way would contribute to solve the above mentioned problem. And we developed a framework to support that process. It complies with ISO/IEC29148 and a combination of several techniques that have been specified in world standards.

1.5 The Structure of This Paper

This paper consists of 6 chapters. The following chapter describes about the related works and a compilation framework that is specified in world standards. In Chapter 3, we explain about the design concept and specification of our framework and its test cases. Chapter 4 discusses about how it contributed in the test cases in the system development practices. And we describe our future work in Chapter 5 and concludes in Chapter 6.

To solve these issues we developed a 2x2 requirement chart to support identifying the fundamental requirements efficiently. Using this chart, we are able to conduct high-quality requirement development on the initial requirements with limited resources and avoid the overrun of cost in the implementation period. Shimazu et al. reported that this chart is able to shorten the project duration and reduce the operational costs. In this paper, we report on its effect in reducing the costs from fixing deficiencies in the implementation period.

Chapter 2

Related Works

2.1 Study of Requirement

2.1.1 Trends in Requirements Engineering

The goal of Requirements engineering is to contribute to the high quality system development by effectively and efficiently identifying requirements. The 3 main communities of requirements engineering are; RE (IEEE International Requirement Engineering), REFSQ (Working Conference on Requirement Engineering: Foundation Software Quality), REJ (Requirement Engineering Journal) and CAiSE (Conference on Advanced Information Systems Engineering). The third community especially focuses on the development of information systems, but the papers are mostly about requirements engineering. Regell et al. [12] illustrated the overview of the discussions in these communities. According to their research, the engineers that determine the requirements take the requirements or demands using some method and provide an organized version of the acquired data as a feedback to stakeholders, which are users and clients. It is safe to say that Participatory Design is being adopted in this method. The methods used to take the requirements and demands are either goal-oriented or aspect-oriented methods, or scenario-based or prototyping methods. All of these methods assume that users do not take any meaningless actions and are always acting with a specific purpose.

2.1.2 Requirements Acquisition from Observation

The field of research where situations in actual sites are observed to produce innovation in engineering is Field Informatics. Field Informatics can be classified into 4 fields: development of descriptive methods (for current situations and events), development of prediction methods, development of designing methods, and the development of transfer methods.

The research on description based on Ethnography has been active since the 1970's [13]. Ethnography originates from the field of anthropology and has evolved in order to describe the behaviors of different ethnicities. Xerox Inc. gathered operators from countries and regions to which the company was planning to introduce the copy machine. The copy machine was placed in an operating room, and the processes were recorded. The methodology adopted in this investigation was Ethnography. As a result, Xerox has successfully provided different UI's with different operability as a component system for many countries and regions. With the advances of recent information technology, investigation of methodology of recording current situations using sensors is starting to attract increased attention.

Participatory design [14], a method to design new artificial objects and social systems, is one of the main research in designing methods. Participatory Design attempts to gather as many participants as possible and form a consensus. Various stakeholders, along with technicians, participate in the system designing process. Workshops are usually held in these system designing methods. These methods are also combined with prediction methods. In other words, the following 3 steps are adopted: (1) extraction of stakeholder models at workshops, (2) visualization of system behavior through simulation, and (3) repetition of consensus formation. These steps are introduced to current system development practices as the basic processes of requirements identification.

2.1.3 Study for improving the requirement quality

In 1980's, software engineering domain started to study how appropriate the system requirements and functional and operational requirements that have been directly proposed by users and sponsors are to achieve the system's ultimate goal [15]. To realize the best performance for users, studies had focused more and more on non-functional requirements [16] and developed methodologies that are represented by the goal-oriented methodology.

Major examples are KAOS, i*Framework [17] and NFR framework [18]. KAOS utilizes four models and allows for step-by-step traceability to specifications and format verification using Temporal Logic. In other words, you could say KAOS is difficult to use for people without such knowledge. i*Framework allows for flexible analysis of dependency relations among actors and could also be used for non-software system development. However it is said that it takes time to understand the meaning of directions of arrows and to master model descriptions. NFR framework decomposes what should be defined as non-functional requirements on the dependency relationship graph divided into three goals. It is rather easy to observe how much the goal is satisfied, but the graph structure sometimes becomes too large [19].

To increase the statement accuracy, Roger presented the importance of the statement elaboration [20].

2.1.4 ISO/IEC29148 and goal-oriented methodology

The requirement engineering standard, ISO/IEC29148, could be considered to embrace the advantage of conventional techniques including goal-oriented methodology. One of the major examples is the decomposition proposed with the goal-oriented methodology that is followed by the logically consistent building blocks in the standard [21]. The "goal" in the goal-oriented methodology is the ultimate stakeholder requirement that is independent of technology and implementation and means the to-be situation (stakeholder requirement concept) after the system is introduced. It also becomes the major information for ConOps together with the building blocks of the whole set of stakeholder requirements.

Meanwhile, the goal is shifted towards system requirements while it is being decomposed. In ISO/IEC29148 [10], processes start by distinguishing stakeholder requirements from system requirements. Then decomposition of the needs and expectations in stakeholder requirements that are independent of technology and implementation iterates. Therefore the stakeholder requirements and system requirements stay separated thus independence (non-technical independence) could be secured. The next process, system requirement development, is conducted by defining the system function and performance to realize the defined stakeholder requirements. Here, quality is defined for each functional requirement. Attributes are described as –ilities: accessibility, compatibility, interoperability, maintainability, probability, recyclability, scalability, testability, securability, usability and more [22].

The problem of description learning of each goal-oriented framework is dealt in ISO/IEC29148. It provides statement structural rules and allows for verifiable description.

Requirement engineering provided by ISO/IEC2914 consists of two processes: the stakeholder requirement definition process and the requirements analysis process. The former process generates the StRS (Stakeholder Requirement Specification) document as the output, and the latter generates the SyRS (System Requirement Specification) document [23]. The most important stakeholder requirement is placed at the top of the building blocks . Stakeholder requirements are described as independent to implementation and in a measurable description in accordance with the structural rules established in ISO/IEC2914. Thus the stakeholder requirements become verifiable and validatable. The highly complete stakeholder requirements are defined and put into the following process. The perfectly traceable system requirements are defined. It is considered that the following processes are expected to be complete with no requirements missed out by using standard techniques such as RTVM (Requirement Traceability and Verification Matrix) [23, 24].

In order to support these processes, ISO/IEC29148 identifies three critical factors; (1) identifying every single use-case and stakeholder that could be assumed in every phase in the system life cycle, (2) defining the boundary by identifying all the operational context of the system, and (3) ensuring the consistency between the two sets of information. [25]

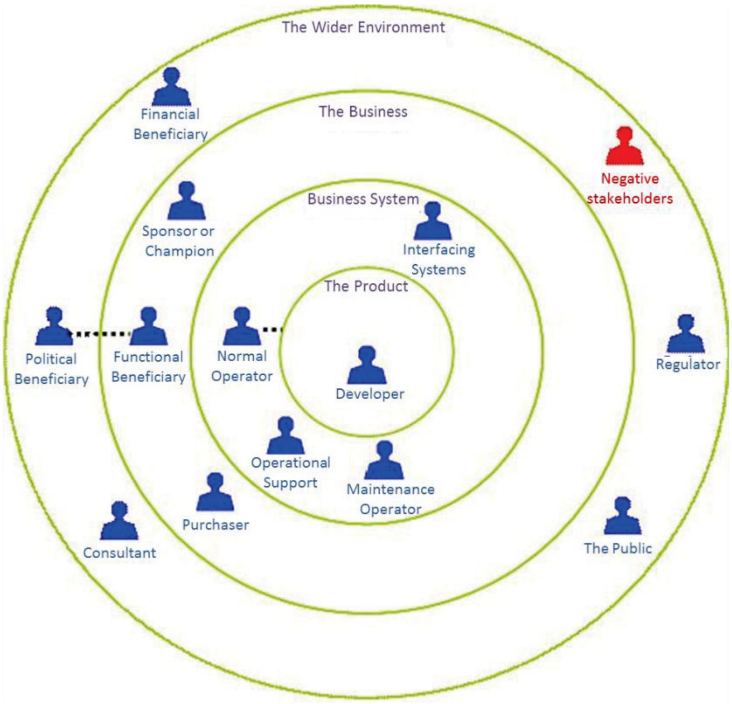
2.1.5 How to identify use-cases

Use-cases are a set of operational instances, in other words, operational scenarios [26] （especially in the model language such as UML and SysML）, and mean process sequences from the activation of the system to the moment when the expected result is acquired. UML and SysML propose a system conditional branching and a description of prior and termination conditions that allow for certain change in abstraction level. Generally use-cases are depicted using use-case diagrams. UML and SysML are the standard AD (Architecture Description) methods [27]. UML is designed for software engineering and SysML for systems engineering.

In UML and SysML, similar to case tools, “actor” is used to describe the subject of the action. What distinguishes these two models is that “actor” is used not only for people but also for other related systems and items. This allows for thorough recognition of all the activities that would affect the system development. However, it does not secure a complete list of stakeholders.

2.1.6 How to Identify the Stakeholders

One of the major techniques to visually identify missed-out stakeholders is Onion Model [28]. This stakeholder model has been widely used in the European business industry and adopted as the standard tool in BABOK（Business Analysis Body Of Knowledge） [29]. To draw this model, you first assume activities from systems development to support operation and then identify stakeholders on concentric circles. A standard set of stakeholders is prepared as a template allowing relatively easy to find missing stakeholders. By placing stakeholders on centric circles considering how much effect they would bring to the system putting ones with more effect on inner circles and ones with less affect (indirect or conditional affect) on outer circles, this model helps you acquire an intuitive understanding and agreement. Stakeholders identified in Onion Model should be individuals or organizations that could be represented by individuals. Stakeholders are connected with arrows defining the direction of influence. The arrows eventually end up in the product or service to be developed, which is described in the center of the circles. Meanwhile, the model does not support any description to present the importance of stakeholders.



The center circle: product or service to develop
The second circle: system (stakeholders who would surely affect in the operational phase)
The third circle: stakeholders who would surely affect in any phase)
The most outer circle: stakeholders who might affect in some phase

Based on [28]

Figure 2.1 Stakeholders in MECE: Onion Model

2.1.7 Context Analysis

The purpose of context analysis is to identify the scope of the achievement of the system's goal that is listed on the top of the

stakeholder requirements described in the building blocks. By conducting this analysis, all the elements such as stakeholders, facilities and even other systems and laws and regulations that would affect the system's intended purpose are identified and the system's entire environment is clarified. Then those elements will be determined whether to be concerned or not. For those determined not to be concerned will be then determined whether to prepare for counter functions or put beyond the scope [30].
Context analysis clarifies what the system is intended to achieve, whom the system is intended for and what the conditions and attributes for the safe use are and what the appropriate physical environment such as facilities and social environment such as infrastructure are.

ISO/IEC15288 as well as ISO/IEC IEC12207 and 29148 that conform thereto define context analysis as follows. All the environmental factors that would possibly affect the system's target operation are defined as "environments". Among the "environments", those any stakeholder wants to take any action are defined as "concerns". In case certain environments turn out to be concerned with specific attributes, such attributes will be identified. In other words, context analysis is about identifying "concerns" out of "environments" and which stakeholders are holding such concerns. Stakeholders are thus thoroughly identified and the system boundary is defined. That's the purpose of context analysis. Therefore the actors described in UML and SysML are considered to be identical to this group of concerns.

Context analysis is generally conducted using context diagrams. Originally, IDEF0 was widely used to describe context diagrams. This method intuitively presents the system's goal as the A-0 (the top) context diagram. After IDEF0, studies revealed that dealing with interrelationships generated by influence as interfaces would allow for clear boundary definition. Since then, DFD (Data flow diagram) and its revised versions have come to be more and more used.

Meanwhile, assumable environmental factors are limitless and such analysis could cost a lot of time. In other words, the time constraint could affect the precision of the analysis. Also there have been no standardized methods proposed so far to describe interrelationships as interfaces. This has often been practiced in one's own way and caused misunderstanding within the project.

2.2 Study of Semantic Model Representation

2.2.1 Entity Relationship Methodology

ER (Entity Relationship) model is a popular data modeling method [31].

From a database design perspective, there are two types of data models: a syntactic model and a semantic model [32]. Syntactic models include the relational model and the network model, while the ER model is a kind of semantic models. Recently, many CASE tools to automatically convert ER-model data to relational-model data have been developed, pursuing a practical database design. This proves the semantic expressiveness of the ER model in representing information structure and its qualification as a framework to share a common understanding of system requirements between users and designers [33].

In [31], Chen captured the relationship between real-world entities in a top-down approach and represented the information structure as an ER model. Later, the concepts of normalization and generalization were introduced [33, 34], as well as a standardized procedure to improve ER schema. In the ER model, information consists of three conceptual components: entity, relationship, and attribute values [33]. The fundamental element of the ER model is an entity, which represents the essential real-world components in the target domain to be modeled, and is particularly unable to be further divided. Relationship relates to more than two entities. "E set" refers to a set of entities of the same type. Similarly, "R set" and "V set" refer to a set of relationships of the same type and a set of attribute values of the same type. Figure 2.2 and Figure 2.3 illustrate a diagrammatic representation of the ER model (ER diagram). In an ER diagram E sets and R sets are represented as rectangles and lozenges. The fact that several E sets are connected by an R set is represented by solid lines connecting each E set and R set.

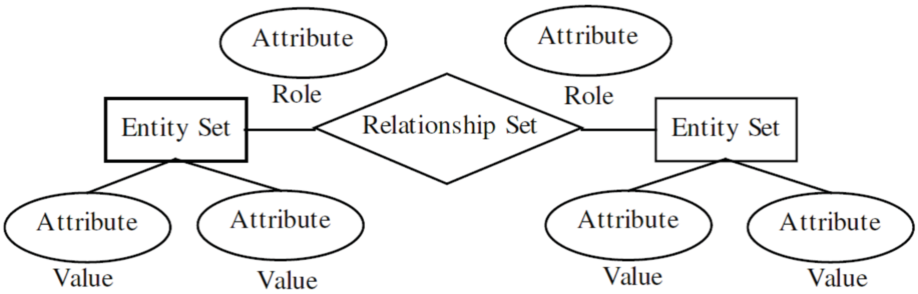
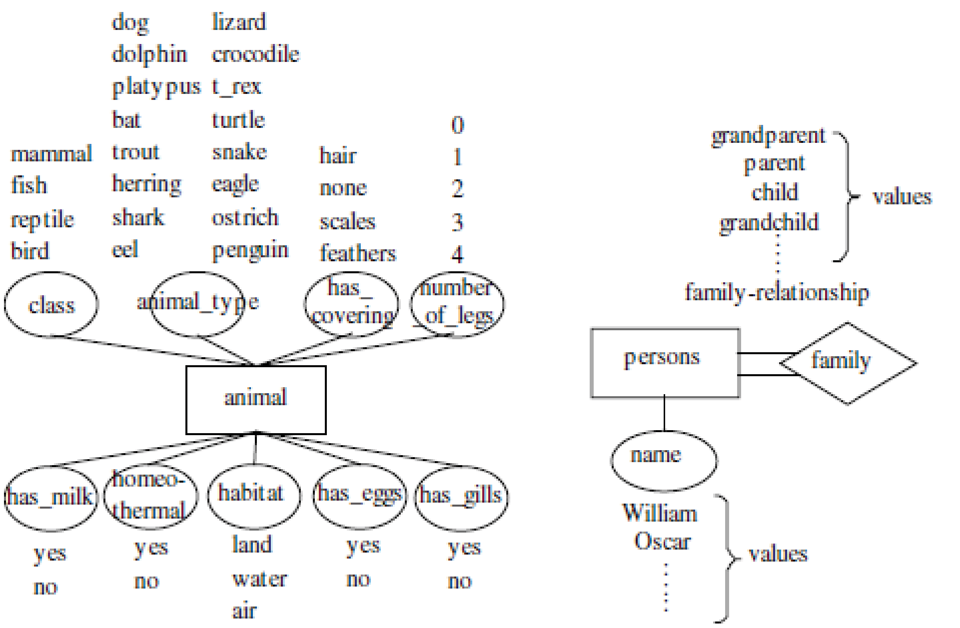


Figure 2.2 ER Diagram: Basic Framework



(a) Animal Classification Database (b) Family Relationship Database

Figure 2.3 ER Diagram: Examples

2.2.2 Refined Entity Relationship Model

Refined Entity Relationship (RER) Model is designed by taking qualitative and structural aspects of databases, by adopting an ER model. The ER model- based design method is likely to be widely applicable to real-world databases with practical design, since it has been proven to be possible to convert an ER model to a network model [35].

In [31], Chen captured the relationship among real-world entities in a top-down approach and represented the information structure as an ER model. Later, the concepts of normalization and generalization were introduced [33, 34, 36], as well as a standardized procedure to improve ER schema. In the ER model, information consists of three conceptual components: entity, relationship, and attribute values [33]. The fundamental element of the ER model is entity, which represents the essential real-world components in the target domain to be modelled, and is particularly unable to be further divided. Relationship relates to more than two entities. "E set" refers to a set of entities of the same type. Similarly, "R set" and "V set" refer to a set of relationships of the same type and a set of attribute values of the same type. Figure 2.2 illustrates a diagrammatic representation of the ER model (ER diagram). In an ER diagram E sets and R sets are represented as rectangles and lozenges. The fact that several E sets are connected by an R set is represented by solid lines connecting each E set and R set.

According to Chen [31], an E set is represented as E ($A1$ /$V1$, $A2$ /$V2$, ... , An /Vn), where E refers to the name (or label) of the E set, and Ai /Vi (i =1,2, … , n) denotes a pair of attribute names (or label) and a V set. An attribute Ai refers to a feature of an E set and is defined as a function from E into Vi. If a minimal subset of an attribute set $X = \{A1, A2, \ldots, Ak\}$ provides a one-to-one mapping from an E set into the direct product of the V set $V1 \times V2 \times \cdots \times Vk$, X is called an identifier of E. The identifier is often underscored in ER diagrams (see Figure 2.2). The notation of an E set is simplified as E ($\underline{A1, A2, \ldots, Ak}$, $Ak+1$, ... , An) [33]. In other words, once the values of the identifiers are specified, those of all other attributes are automatically fixed[1].

Similarly, an R set relates several E sets $E1$, $E2$, • • •, Em. A set of mutually related entities ($e1$, $e2$, ... ,em), where $ei \in Ei$ (i =1,2,...,m), is called a relation and an R set is a set of relations with the same type. An R set is denoted as R ($E1$ / $L1$, $E2$ / $L2$,..., Em / Lm: $A1$ /$V1$, $A2$ /$V2$, ... , An / Vn), where R denotes the name (or label) of an R set, Li denotes the role of Ei in an R set R, and Ei / Li (i =1,2, ... , m) denotes a pair of Ei and Li.

---

[1] If every non-identifier attribute in an E set is functionally independent of other non-identifier attributes and is fully dependent on the identifiers of the E set in an ER model, the model is "normalized" [33, 34]. In this paper, we assume that ER models are normalized.

Some R sets include pairs of an attribute and a V set *Ai* /*Vi* (*i* =1,2, ... , *n*). Similar to E sets, the notation of R sets is simplified as *R* (*E1*, *E2*, ... , *Em* : *A1*, *A2*, ... , *An*) [33].

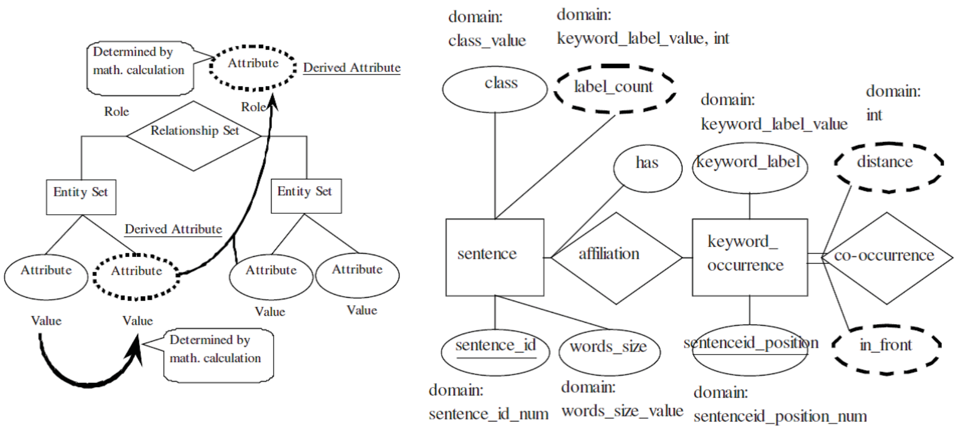
There wasn't standardised methodology for structuring information to deal with systematically, before ER model was developed. This model has been used for long period for building relational databases. When a relational database is generated from an ER model, the number of tables in the database is equal to $|E| + |R|$, where $|E|$ and $|R|$ denote the numbers of E sets and R sets. In practice, when the database is implemented in a DBMS (DataBase Management System), tables of the relevant number are further generated through generalization, instantiation, and unification operations.

On the other hand, ER model presentation is not enough for multiple tables to structure whole information systematically. The animal classification database in Figure 2.2 (a) is represented without redundancy by a single table. The size of its single table representation may increase up to infinity. These facts are more intuitively understood with the corresponding ER diagrams. Figure 2.3 indicates the ER diagrams. Note that the "family" relation is a binary relation between two persons, which is represented by a double link connecting them. An example of an RER diagram appears in Section 3.2 (Figure 2.4). If a database's normalized representation is a single entity, like the animal classification database, it can be represented as a single table and propositional logic learners are suitable for them. On the other hand, if a database's normalized representation includes relationships, like the family relationship database, one can obtain meaningful results only by employing predicate logic learners.

RER (model identifies specific attributes that are functionally dependent on other attributes, or whose values can be calculated with values of other attributes (and themselves). As mentioned in the previous section, values of attributes are determined by the values of the identifiers. Among them, we term those attributes whose functional dependency is defined as functional expressions "derived attributes."

A normalized form and an example of the RER model are shown in Figure 2.4. The left-hand side is the normalized form and the right-hand side shows the RER Diagram of "Responses" database, which will be utilized later in Section 6. In this example, "label_count" attribute and the "co-occurrence" attributes ("distance" and "in_front") are derived attributes.

The redundant attributes caused by functional dependency, introduced in the Extended Entity-Relationship (EER) model, are considered to be equivalent to derived attributes in this paper. However, their purpose, which is to eliminate such redundancy through normalization, differs from our position to utilize them for advantageous purposes.



(a) Normalized Form (b) RER Diagram of "Responses" Database

Figure 2.4 Normalized Form and Example of RER Model

Meanwhile, the database for family relationship rule induction is properly represented only by multiple tables.

We suppose that the adequacy (the effectiveness of the result) of a machine learning system to a target domain can be estimated with the number of optimal tables for the domain. Specifically, ILP-based learning is appropriate for a domain if the difference between the redundancy (i.e., the number of missing data) of a single table representation of the domain and that of a multiple table representation is significant.

In such situations, the corresponding ER diagrams include relationships among multiple entities, which ILP systems are effective in handling. Among the PROGOL example input files [37], the animal classification database is represented without redundancy by a single table. Meanwhile, the database for family relationship rule induction is properly represented only by multiple tables.

2.2.3 Object Process Methodology

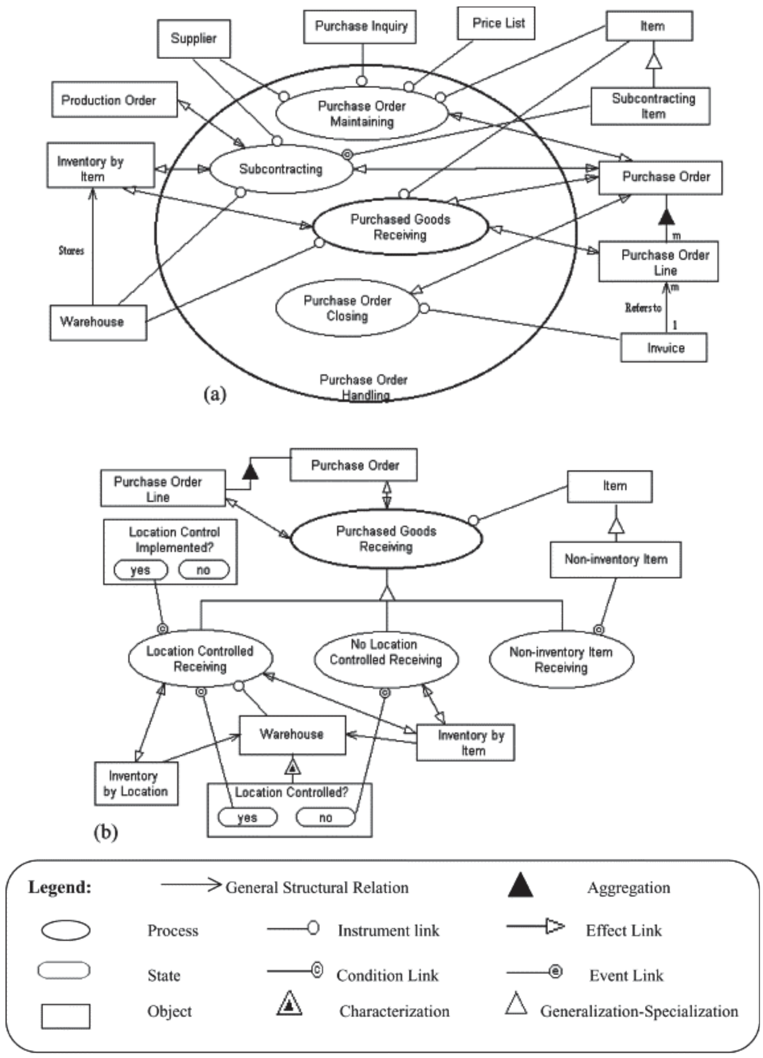
OPM (Object Process Methodology) is a model language, which utilizes the unified description rules independent of the abstraction level [38] (Figure 2.5). OPM consists of OPD（Object Process Diagram）and OPL (Object Process Language). OPD shows the object model and the process model being connected to each other with a line segment. Physical entities and information are described as "entities". These two kinds of entities are described distinctively to each other. Physical objects include individuals. Entities are categorized in objects, states or processes. Objects exist during the certain period of time. States are the situation the objects could be in. Processes generate one or more changes in objects (change in state, consumption, deletion or transformation). In OPM, these objects are connected with links. Some links mean structures, and others, processes. The former links depict the interrelationships between two objects and will never be deleted. The latter links connect process and object or process and process. When a process and an object are connected, the process will change the object state. When two processes are connected, the output of the first process will be handed over to the next process as its input.

DFD and ER (Entity Relationship) are currently used in systems engineering. But they have been developed originally for software development. You could say that OPM is an integrated modelling language combining the two description methods.

What is distinct from DFD and ER is that OPD result could be translated to OPL right away, which is readable (Figure 2.6). This would support system engineers to judge whether the modelling is depicted as intended.

For practical use, tools are being developed to transform the OPM result into UML [39].

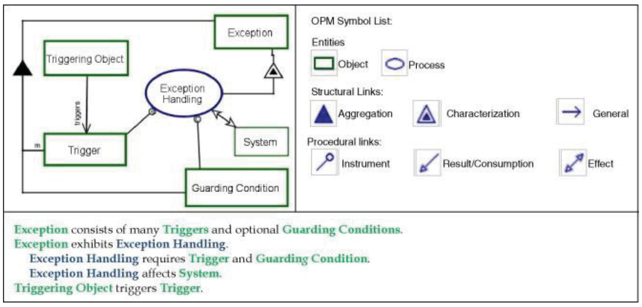
OPM is now expected to become the basis of an emerging ISO Standard for Model-Based Standards Authoring, by ISO TC 184/SC 5 OPM Study Group.



(a) Purchase phase in the operational stage in the system life cycle.
(b) Product acceptance phase in the operational stage in the system life cycle.
Shown in the rounded square is part of the legend.

Based on [44]

Figure 2.5 Two Examples of Modelling in OPD



Example of OPD modeling (upper left)
Part of description rules for OPD (upper right)
OPL Automatically converted OPL (bottom)

Based on [44]

Figure 2.6 An Example of OPL

2.2.4 Requirement Engineering Education for non Experimented Students

Recently dramatic structural reforms have been reported in the industrial world almost on a daily basis, since the advent of the Internet. Alliance or merger between corporations, which were thought to be in different industries few years ago, is no longer a rare event. Similar changes are required in University education.

In the past, the courses at universities were different between majors, and different classes were prepared for each of the courses. It was especially uncommon to have a cross-disciplinary program between management and engineering. The reason was that these fields were considered to be fundamentally different. However, in real life, it has become more difficult to expect a breakthrough (fundamental solution to complex issues or development of products for the future generation) from conventional ways of operation, which focused on issues within each field of expertise or organizational structure. In many cases that have achieved breakthroughs, a project-type organization is being utilized. In these cases, experts from various fields have exchanged opinions and ideas from various viewpoints, investigated their practicality and the risks involved, in a short period of time.

University education is expected to enhance the ability of the students to thoroughly consider what kinds of system the market or the society requires. Previously in engineering, “requirements” had to be identified by the supposed users and sponsors. “Requirements” are functions and performance of the system that has to be implemented and methods that achieve them. “Requirements” are organized so that there is no contradiction among the needs and demands of the supposed users and sponsors. Currently, “requirements” leads systems engineering and are developed as specifications upon which all the stakeholders can agree. The systems engineering education program at universities also needs to meet these social demands. Specifically, the program should motivate students majoring in engineering to cooperate with students majoring in management and to acquire skills to solve problems together with them. They also should acquire the skills to accurately understand the current market and social situations and elicitate potential needs and demands and convert them into explicit “requirements”.

2.3 Paradigm Shift of Engineering Education

Froyd et al. investigated the transition of education methodology in engineering universities during the past 100 years [40]. According to the investigation, there are 5 major paradigm shifts, which were triggered by social background. The first paradigm shift occurred during the period around the Second World War. This is the period of transition from a style focusing on practice to a type utilizing scientific analysis. During this period, mathematical models and theory-based approaches had been introduced to engineering classes. This coincides with the period during which scientific approaches had been introduced in the industrial world to improve management quality. The second shift was triggered when the predecessor of the Accreditation Board for Engineering and Technology (ABET) decided to add quality control to the engineering education curriculum. Specifically, students were educated to consider what kind of results can be expected from the current technological development operations that they are working on. From late 1980’s to 1990, this style has become common in engineering education at universities in the United States.

The third occurred when a new curriculum teaching the importance of design was introduced to engineering education. The reason behind it was that multiple reports were released around this period, which pointed out the tendency that education at universities overlooked the overall design. By 2005, more than half of faculties at engineering universities in the United States adopted design-oriented education. At first, most of the bachelor students did not understand the essence of design. However, they did acquire the knowledge that it leads to innovative ideas during this type of education. The fourth paradigm shift occurred when sociological approaches were introduced in the late 2000's. The goal of this shift is to provide knowledge and experience for contribution to society. In particular, inquiry-based learning methods, which include problem-based learning and project-based learning, were adopted. The importance of so-called active learning, which involves participation of both the lecturer and the students and differs completely from lecture style classes, was claimed by many studies. The last shift is the introduction of social information infrastructure to university classes. This gradually occurred from the late 1950's to the early 2000's. The material is gradually shifting from books to video contents. Currently, there are even remote classes using the internet. Previously the classes had taken the form of one lecturer providing the same contents to a group of students. It is shifting towards unique instructions for each individual. At the same time, introduction of an intelligent tutor system, or game-like elements focusing on development of competitiveness are starting to happen.

It's believed that the first four paradigm shifts are not changed but rather additions to the required elements in engineering education. In other words, classes focusing on practical training are essential to engineering education. We also believe that classes that can provide the knowledge and experience required to think about what the optimal product is, how to develop a design that accurately reflects the ideas, and how to cooperate with students from non-engineering fields.

2.3.1 Study of Hands-on Education for Systems Engineering

Research on hands-on type curricula applicable to various fields of engineering has been conducted by Bonnema et al, Castles et al and

Yanfei et al [41, 42, 43]. However, they focus only on the methodology of manufacturing actual products and do not include concept development and requirement development processes of the projects. Schilling et al and Pomalaza-Raez et al have also reported on this type of classes [44, 45]. These include the processes of concept development and requirement development, but are applied only in the field of aeronautics. We propose a novel hands-on type curriculum that includes both concept development process and the requirement development processes. Our curriculum can also be applied to education in various fields of the engineering domain. Hole introduced the IBM case study on stakeholder requirement [46]. Interactive Solution Marketplace (ISM) in IBM is a single point of entry on the ibm.com website for browsing and searching for a suite of solutions as opposed to individual software and hardware items. ISM instructs the most important activity of systems engineering; stakeholder requirement.

Consensus on core stakeholder requirements was achieved early during project development, and the technical reviews allowed the group to identify and resolve key issues before proceeding into subsequent phases. Lande was interested in the maturation of master's students with backgrounds in mechanical engineering adjusting to a project-based learning experience centered on the design thinking methodology and processes [47]. He used a combination of theoretical approaches for design research, engineering education and the learning sciences. At the University of California at Berkeley, embedded systems have become a traditional area of strength in the research agenda [48]. In parallel to this effort, a pattern of graduate and undergraduate courses has emerged that is the result of a distillation process of the research results. Sangiovanni-Vincentelli presented the considerations that are driving our curriculum development and we review our undergraduate and graduate program. In particular, we describe in detail a graduate course (EECS249: Design of Embedded Systems: Modeling, Validation and Synthesis) that has been taught for six years.

A common feature of the education agenda is the search for the fundamentals of embedded system science rather than embedded system design techniques, an approach that today is rather unique. The three technical universities in the Netherlands (Eindhoven University of Technology, Delft University of Technology and University of Twente), abbreviated as 3TU, started a joint master on Embedded Systems in 2006

[6]. Embedded Systems is an interdisciplinary area of Electrical Engineering, Computer Science, Mechanical Engineering and Applied Mathematics. They studied the background of the master and presented the curriculum of the masters at the three sites. The Plessey Telecommunications Company and Loughborough University disrupted the students' software development progress.

These actions appear mean and vindictive, and are labeled 'dirty tricks' in their words but their value was appreciated by both the students and their employers. The experiences and learning provided by twenty 'dirty tricks' are described and their contribution towards teaching essential workplace skills was identified. The feedback from both students and employers were mostly informal but the universally favorable comments received give strong indications that the courses achieved their aim of preparing the students for the workplace. They identify some limitations on the number and types of 'dirty tricks' that can be employed at a university and concludes that companies would benefit if such dirty tricks were employed in company graduate induction programmers as well as in university courses.

Forsberg et al. claim that education of systems engineering should be conducted with the involvement of the students from both the departments of social science and departments of science and engineering. They also claim that systems engineering is not efficiently and effectively taught at universities due to the organizational structure. Departments and graduate schools are independently structured, and thus have independent curricula. They especially stress the drawbacks of separating MBA, which focuses on management, and engineering departments. However, universities and graduate schools which claim to have adopted systems engineering from Europe and the US, courses of systems engineering practices can only be taken by students who have already taken courses from the engineering departments. There are a number of prerequisite courses, which also limits the students to those from the engineering departments.

On the other hand, there are hardly any studies that have introduced this type of class other than the one conducted by shimazu et al. [50]. In this paper, we report on the effects that the class proposed by shimazu et al. had on the factors mentioned in the fourth paradigm shift.

2.4 Challenge in stakeholder requirement definition

Given what has been described in this chapter, we could conclude that no generic method has been proposed to systematically and seamlessly generate a list of stakeholder requirements out of stakeholder needs that would be put into the following process where the optimal system requirements are generated (that could be traced back to the stakeholder requirements).

The purpose of both the context analysis and the use-case identification is to generate stakeholder requirements. They are handling the same information in different contexts. Those terms and information are likely to be transferred without clarifying ambiguities. For example, according to ISO/IEC29148, each stakeholder requirement shall be able to trace back to specific stakeholder. Meanwhile, these requirements are described in well-informed sentences under the structural rules (Section 2.2). The subjects of such sentences are not always the owners of the requirements. Sometimes they are one of actors shown in the use-case diagram. These actors are identical to concerns of environments in the context analysis. It is considered that the context analysis and the use-case identification does not support sufficient enough to identify a highly complete list of stakeholders without leaving out any important stakeholders. Missing out such stakeholders could lead to a fatal defect of the system. Therefore, in many cases the use-case identification and the context analysis are conducted to some extent to intuitively confirm that they are congruent and then stakeholder requirements are to be analysed.

ISO/IEC15288, which underlies ISO/IEC29148, is the Vee modelled system development, where parallel effort of off-core and on-core practices are important [50]. Stakeholder requirements are defined independently of implementation or technology, while taking in candidates and feasibility of implementation at the same time. That is considered to be making this complicated problem.

Such stakeholder requirements are handed out to the next process as an input to generate system requirements. It is considered to be the cause of groundless specifications.

Chapter 3

Requirement Development Framework designed by Synergistic Models

3.1 Challenges in Stakeholder Requirement Definition

Given what has been described in this chapter, we could conclude that no generic method has been proposed to systematically and seamlessly generate a list of stakeholder requirements out of stakeholder needs that would be put into the following process where the optimal system requirements are generated (that could be traced back to the stakeholder requirements).

The purpose of both the context analysis and the use-case identification is to generate stakeholder requirements. They are handling the same information in different contexts. Terms and information are likely to be transferred without clarifying ambiguities. For example, according to ISO/IEC29148, each stakeholder requirement shall be able to trace back to specific stakeholder. Meanwhile, these requirements are described in well-informed sentences under the structural rules (Section 2.2). The subjects of such sentences are not always the owners of the requirements. Sometimes they are one of actors shown in the use-case diagram. These actors are identical to concerns of environments in the context analysis. It is considered that the context analysis and the use-case identification does not support sufficient enough to identify a highly complete list of stakeholders without leaving out any important stakeholders. Missing out such stakeholders could lead to a fatal defect of the system. Therefore, in many cases the use-case identification and the context analysis are conducted to some extent to intuitively confirm that they are congruent and then stakeholder requirements are to be analysed.

ISO/IEC15288, which underlies ISO/IEC29148, is the Vee modelled system development, where parallel effort of “off-core” and “on-core” practices are important [50]. Stakeholder requirements are defined independently of implementation or technology, while taking in candidates and feasibility of implementation at the same time. That is considered to be making this complicated problem.

Such stakeholder requirements are handed out to the next process as an input to generate system requirements. It is considered to be the cause of groundless specifications.

3.2 Requirement Model

Based on a study outcome using actual requirement statements from 2005 to 2007 [51], a requirement model proposed in 2009 (Figure 3.1). In this paper, the technical term model is employed in reference to Forsberg [52]; it focuses attention on critical issues and retains the essence of the requirement types. In this case, the requirement model is utilized as a template for requirement design. OPM (object-process methodology) is employed in this paper as the presentation method for this model. OPM is an effective tool for the authoring of systems engineering outputs.

There are three types of requirements in general: operation, functioning and maintaining. The most significant requirement is operational, which specifies the “as is” status and the ideal “to be” status. This transformation is realized by the functional requirement. The “to be” situation is sustained by maintaining the requirement. These days, many initial requirement documents include technical issues concerning the implementation of the requirements. This should not be discussed in this step, but should be discussed at the time of delving into each functional requirement.

Requirement in systems engineering is discussed from various viewpoints. Most of the requirements can be characterized by two axes: x axis according to development and use, and Y axis according to key enabler and objectives (Figure 3.1).

For example, most initial requirements describe the technology or methodology from the viewpoint of X1. This is due to the fact that with the advent of the Internet, general users now have access to cutting-edge technology, and they expect newer technologies to be incorporated. This situation leads to discussions on requirements specifications based on technological issues. To avoid problems that could alter the essential aspects of services in the Development Stage, we need to clearly classify the “as is” and the “to be” state as information in ConOps. In other words, this is equivalent to the dimension (i), the point at which Y2 and X2 intersect. Therefore stakeholders can determine the technology or methodology to be used.

Currently, we are not able to allocate sufficient resources to requirements development, and the statements are deployed into service requirements without developing ConOps. It is easy to imagine that the resulting information service systems do not realize user scenarios. Looking at the results of various projects, we are able to claim that this is true to many of them [53].

Engineers classify the requirements according to the 2x2 requirement chart. This enables users to notice whether an operational requirement needs to be generated. We describe how to use the 2x2 requirement chart (Figure 3.2)

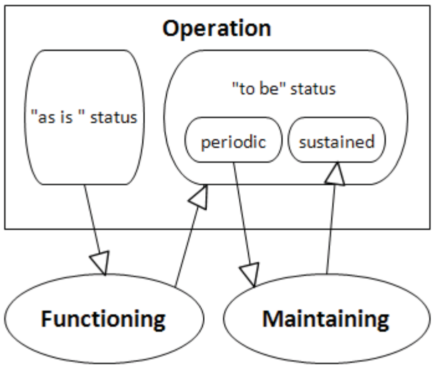


Figure 3.1 Requirement Model presented in OPM

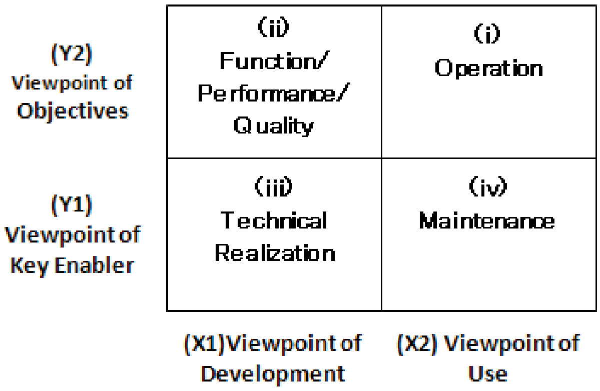


Figure 3.2 2x2 Requirement Chart

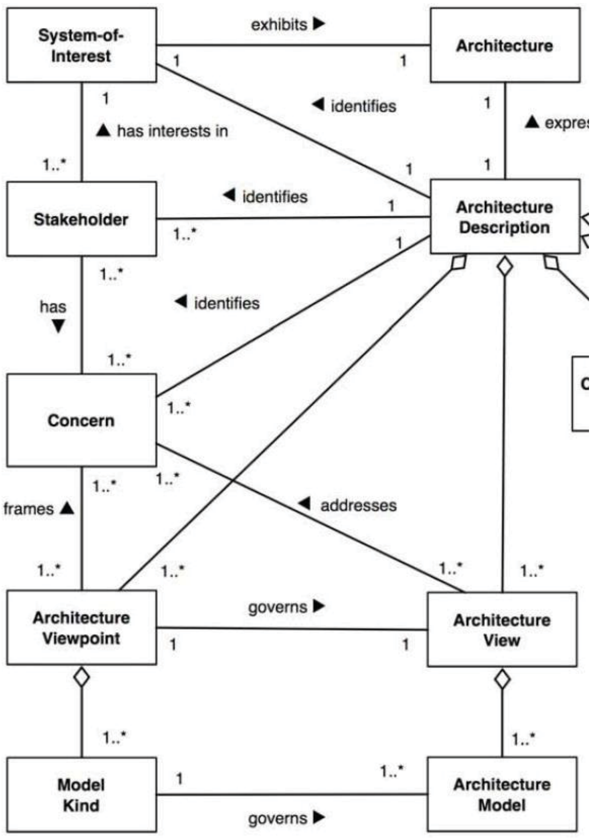
3.3 Contribution of 2x2 Requirement Chart

2x2 requirement chart is a requirement development framework that enables systematic generation of StRS out of stakeholder needs. StRS will then be put into the subsequent process to generate SyRS (that could be traced back to StRS.)

Being conforming to ISO/IEC2914, this framework could support the goal-oriented methodology and bring about efficient elaboration of statement that Roger proposes.

The important information described in SyRS is the system architecture. Generally it is depicted in AD (Architecture description) including UML and SysML [54]. To avoid missing out anything, it is suggested that multiple architectures should be prepared and they should be depicted in multiple views (Figure 3.3: excerpt from [55]). It is generally regarded that complete system design requires at least three architectures each with Operational view, Functional view and Physical view. Our 2x2 requirement chart is considered to be able to contribute to the system architecture with the Operational view since it contributes in writing ConOps.

Expected contributions are: (1) defining logical and consistent stakeholder requirements with complete traceability from needs, and (2) allowing identifying system requirements that would satisfy each statement. Thus it is considered that the more demanded functions would be identified.



Excerpt from [55]

Figure 3.3 Stakeholder, Architecture and View

3.4 2x2 Requirement Chart for Discriminating Stakeholder Requirements from Ambiguous information

The basic way to use the 2x2 requirement chart is shown in (Figure 3.4). The processes are conducted as a collaboration between services engineers and stakeholders [56].

1) A statement from initial requirement labeled as STM_01
2) STM_01 is converted into engineering terms as STM_01_en
3) STM_01_en is classified as an appropriate dimension.
4-1-1) if STM_01_en is classified as ii or iii, this statement is identified an operational requirement is generated as STM_01_en_01.
4-1-2) STM_01_en_01 is converted into engineering terms, step 2.
4-2-1) if STM_01_en is not classified as ii or iii (is classified as i or iv), utilize it in ConOps
4-2-2) End

We consider the project of developing a flying vehicle for one person. The initial requirement states that vehicle is "easy to handle." Converting this into engineering terms, (a) "The Length shall be under 3meters." and (b) "Genetic algorithm shall be run for selecting flight path activated" would be obtained. We describe how to enhance the requirements specification based on these statements using the 2x2 requirement chart.

Statement (a) belongs to (ii) in the chart (Figure 3.5). We then discuss why this statement is required and produce one or more statements from it. As a result, "2 healthy men shall be able to carry without external power" and "the system shall be stored indoors when not in operation", which belong to (i) and (iv) respectively, would be identified.

Statement (b) belongs to (iii) in the chart (Figure 3.6). Conducting the same operation as (a), we obtain two statements "Operation shall be in either manual or automatic mode at the time of flight path selection."

The main goal of our proposal is to lead the requirement statement processing operation (which tends to lean towards technological issues), to statement processing (which considers users and the user scenarios). In the actual requirement development operation, we also consider the statements in (ii) and (iii) if needed.

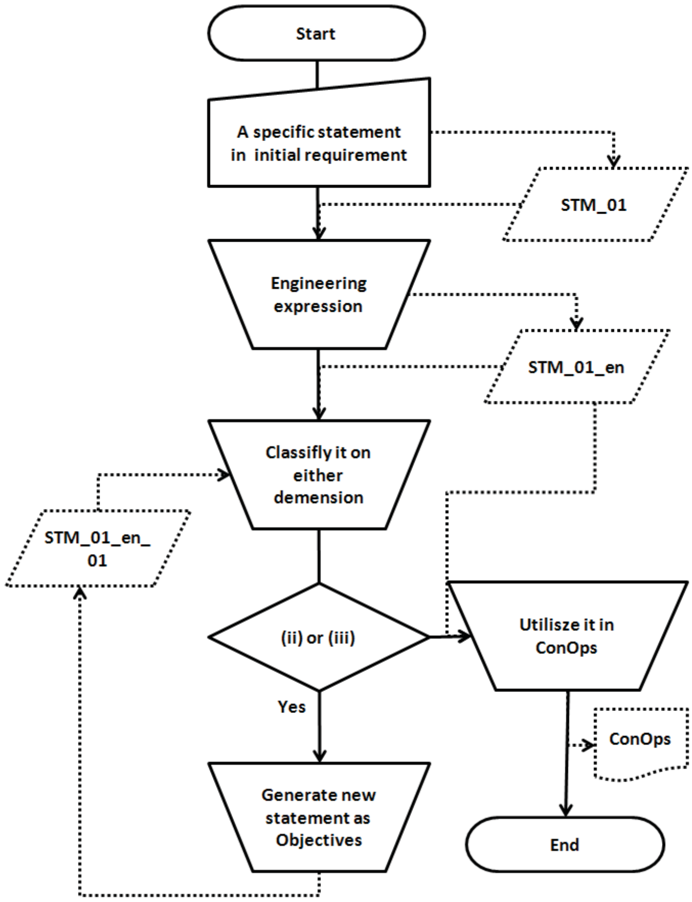


Figure 3.4 2x2 Requirement Chart Utilization Process

	(X1) Viewpoint of Development	(X2) Viewpoint of Use
(Y2) Viewpoint of Objectives	Length shall be under 3 meters.	The product shall be moved by 2 male adults without using external power.
(Y1) Viewpoint of Key Enabler	An axis shall be made from 2.5 m seamless wooden board .	The product shall be indoors when not being operated.

Figure 3.5 Example of Applying the Chart (1)

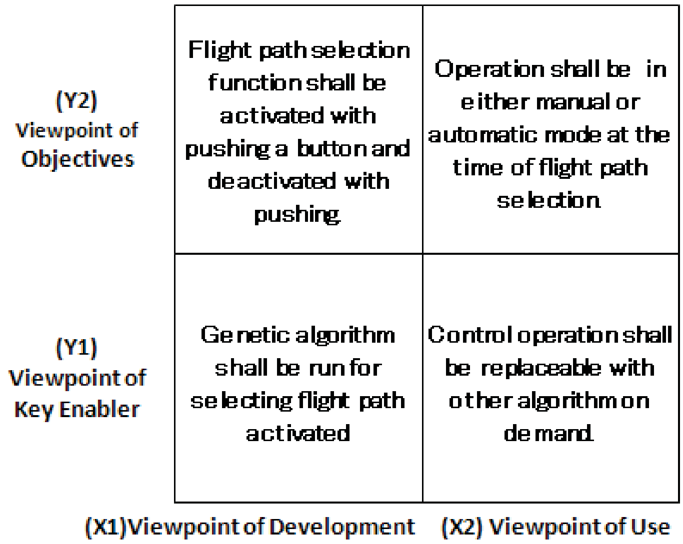


Figure 3.6 Example of Applying the Chart (2)

3.5 Synergistic Model-Based requirements definition framework

3.5.1 Concept : Combining Onion model and OPL to describe use-cases and contexts at the same time

In order to solve the challenges mentioned in Section 2.7, we tried to develop an effective tool by optimally combining what had previously proposed in the related works. Those previous study results have been put into practical use and found to be effective in many cases.

We developed a method conducts these processes systematic by maintaining them holistic and traceable. In this method, not only "people", but also systems, products, tools, parts and environment, which either directly or indirectly affect the system, are added to the list of stakeholders. In other words, the stakeholders are an extension of the actors in UML, and indicates all the entities that produce some context for the context analysis in the systems engineering standards.

The main characteristic of the proposed method is that it prevents users from overlooking information and elements within each process by combining multiple methods from the standards and previous methods.

What was impossible to achieve with the introduction of the previous methods are stated in the related work section. Furthermore, because the results are in the form of diagrams and text, obtaining a consensus among various stakeholders becomes more efficient.

The use-case diagram described in basic flows is easy to understand for non-technical stakeholders. Each use-case is a set of instances of operational scenarios. So if you lower the abstraction level, you could identify operational scenarios seamlessly. In this process, implementation is assumed to some extent and the V-model-conformed practices are conducted. The context analysis is about defining the external interfaces, but it could be about defining stakeholder requirements in a different viewpoint. Challenges in linking these two practice results are (a) completeness of the list of stakeholders and (b) development of integrated description. Considering that Onion Model would solve (a) and OPM (b), we designed a framework as described in the following section.

3.5.2 Design of Outline：Combination of Onion model and OPL to describe use-cases and contexts at the same time

Figure 3.7 is the RER (Refined Entity Relationship) diagram showing output information derived from each stakeholder requirement development technique and their relationships [57]. Actors correctly defined in the Basic flow use-case directly affect the functions and performances of the system. They are identical to the concerns shown in the A-0 diagram if it is correctly analyzed. They are also identical to the complete set of stakeholders and part of the related systems and products allocated in Onion Model.

Actors in use-cases other than Basic flows (extend flows, include flaws and alternate flows) are expected to identify concerns in the context analysis results described in diagrams A-1 to A-n. The set of actors or concerns are identical to the set of stakeholders and related systems and products shown in Onion Model. To deliver successful development, Onion Model prepares a template to confirm whether the diagram is complete with no missed out the elements.

The actors and the Basic flows and their pre-conditions and post-conditions in the use-case identification results are included in the Level-0 OPM diagram. Similarly, the concerns, output(s) and control(s) in the context analysis result shown in A-0 are included in the Level-0 OPM. Therefore, creating a Level-0 OPM diagram means defining Basic flaws in the use-case and A-0 state of the context.

Figure 3.7 shows the top level (Level-0) of description to avoid complexity. In further levels (Level-1 and further) of OPM diagrams, remaining attribute values such as alternate flow(s) in the use-cases and the results of Leve-1 and further diagrams can be described at the same time.

In order to completely identify the stakeholders and the related systems and products, we propose a framework to identify use-cases by using Onion Model as its base and adding context description using OPL, the OPM's language description.

By using the Onion Model template, a basic set of stakeholders will be quickly identified and the stakeholders will be connected to the service/product to be developed. Being compliant with the description rules, an object (stakeholder) and another object (stakeholder or a physical object or information) will be connected via a process so that no interfaces will be left out. OPM depicts the structure by connecting objects with lines. The structure is easily depicted in different abstraction levels. This makes it easy to find any missing stakeholders. So we considered that using this framework would systematically (to some extent) ensure consistency between the results of use-case identification and context analysis.

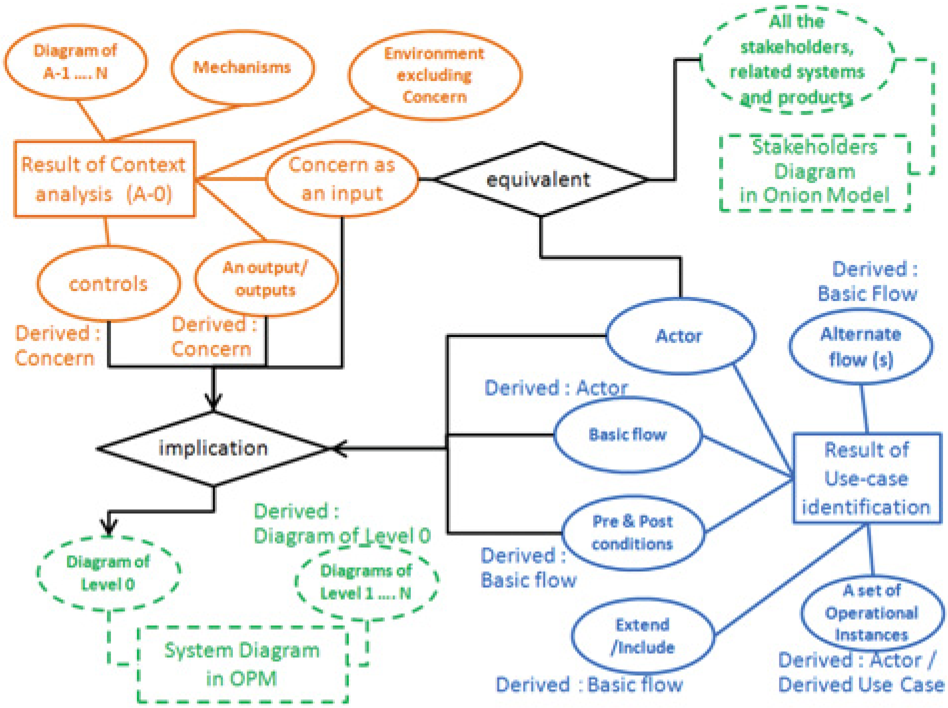


Figure 3.7 Stakeholder Requirement Development Models and their Relationships

3.5.3 Detail Design: OPM to UML/ SysML use-case diagram

OPM and UML/SysML exhibit various benefits, so ways to create synergies have been considered, especially in two issues. The fist issues is that the result of OPM is able to be mapped to UML/SysML concepts and symbols. The other is it's easy to translate automatically OPM to UML/SysML.

OPM and UML/SysML are respectively evaluated as follows;

OPM, especially OPD, is well-organized diagrams hierarchy promotes holistic understanding of the system and its environment. The Emphasis is on simplicity, compactness, and minimalism. OPM has two representations, graphic and text. On the other hand, it's not easy to express finer points of complex systems. In other words, many engineers want to have more suitable for describing the overall picture of the system and its hierarchy levels, activities which are typical of the early stages of the conceptual design.

In contrast, UML/SysML take rich and comprehensive language constructs. Therefore, Different characteristics of the system can be expressed using the available variety of diagrams, including requirements and engineering analysis. The most significant point is these are standards with common and widely supported notation, promotes interoperability. On the other hand, Diagrams tend to be complicated, the underlying concepts may be spread over different views. In other words, many engineers want to have More appropriate when detailed views are required, usually arise in the later stages of the design process.

The mapping from OPM to UML/SysML is “one to many.” In more detail, a single OPM diagram type (OPD) translates to several SysML elements in different SysML diagram types. The mapping also context-sensitive as SysML is ontologically overloaded in few concepts. For instance, an OPM Process is mapped to Use Case (in a Use Case Diagram), Operation of a block (in a Block Definition Diagram), Action (in an Activity Diagram), State transition Trigger (in a State Machine Diagram), Message (in a Sequence Diagram)

There isn’t a single global mapping table between OPM and UML/SysML. For each target UML/SysML view (diagram type) there is a designated OPM-to-SysML mapping scheme and translation algorithm. Mapping schemes and translation algorithms can be systematically developed for use-case diagram, block definition diagram, activity diagram, state machine diagram, sequence diagram and requirement diagram. In this study, the main issue is developing seamless and a systematic mapping framework between context diagram. And use-case, in OPM, Therefore, how to map OPM to use-case diagram is focused on.

Table 3.1 shows mapping rule among OPM, UML/SysML and Context diagram. Use Case Diagram consists (mainly) of Actors, Use Cases and the relationships among them. Use Case Diagram and context diagram are created from OPM is based on Environmental Objects (the actors) and the Processes (use cases) linked to them.

Table 3.1 Mapping among OPM, UML/SysML and Context Diagram

OPM	UML/SysML	Context Diagram
Environmental Object Object	**Actor** ActorName	**Concern**
Systemic Process connected with Environmental Object Object — Processing	**Use Case** UseCaseName	**Interface (to product / system)**
Generalization-Specialization relation between Objects or between Processes	**Generalization** of Actors or Use Cases	**Interrelationship**
Exception Link; **Invocation Link**	Use Case **«extend»** relationship «extend»	**Mode transition**
Process **In-zooming**; **Aggregation-Participation** relation	Use Case **«include»** relationship «include»	**Mode transition**
Any type of **Procedural Link** between an Agent ("actor") Object and a "use case" Process	**Association** between Actor and the corresponding Use Case	**Effection**

3.5.4 Processes : Combination of Onion model and OPL to describe use-cases and contexts at the same time

The framework for generating stakeholder requirements consists of 3 sub-processes during the stakeholder requirement definition process and requirements analysis process. One is a Concept Operation Definition Process, and the other is a Use Case Identification Process.

i. Concept Operation Definition Process

Input: text describing the intention and objective of systems development and problem solving, information required to identify the output.

Output: (1) All Modes that constitute the Operation Stage, (2) All phases that constitute systems Lifecycle and all actors appearing in each phase, (3) effects each actor has on the system in each mode, effects and contributions of the system towards each Actor, expressed in the structured text format.

Activity: Place all actors on the Onion Model developed by Ian Alexander. Using OPM and OPD for presentation, all the actors are connected to the products inside the center circle, the system. This is called "onion model in OPM (OMO)." OMO is prepared for each mode. For identification of Moses, we adopt "The 6Ms", a standard classification for cause and effect diagrams. The elements on the axis of measurement are the modes. We consider elements on the Man, Environment, and Material axes as objects and elements on the methods axis as processes in OPM, which are placed on the onion model and are connected using OPD. When an arbitrary actor is not connected to the product at the center, it means that something has been overlooked in the process (element of Method). On the contrary, if the process exists but there is no actor that activates the process, either Man, Environment, or Material is missing. When there are elements that appear in the cause and effect diagram but cannot be expressed in the onion model in OPM, mode (in other words, Measurement) is missing. We create a situation where all the elements on the axis of the cause and effect diagram and the descriptions on OMO can be perfectly traced. Finally OPD is uniquely converted to OPL.

ii. Use Case Identification Process

Input: (1-1) All modes constituting the Operation Stage. (1-2) All phases constituting the systems Lifecycle and all actors appearing in each phase. (1-3) Effects each actor has on the system in each mode, identified effects the system has on each actor.

Output: (2-1) Use case consisting of entities, actions and measured values. (2-2) Mode that the output use case appears in, and identification of whether it is negative or positive. (2-3) derivation of constraint requirement.

Activity: We reference the output results (1-3), set the Object as the subject, Process as the predicate, and the target Object as object, and convert them into use-cases. When an object has not stated and the returned value cannot be identified, we identify the value instead of the object that the process, activated by the initial object, targets. The object and the process connected by a condition line are identified as a constant requirement that has no measurable return value. We also identify relations connected by Exhibition, Generalization, and Instantiation lines as constraint requirements.

In this process, we check the traceability by checking if all the objects, processes and lines in OMO appear as the main text of use cases, and if all the modes as conditions that validate the main text.

3.6 Example result of framework utilization

Figure 3.8 shows the stakeholder requirement definition process for a Lawn care system using the developed framework. The life cycle stage is “Utilization” and the mode is “in use”. The Lawn care achieves its targeted goal by an operator manicuring the lawn. Manicuring the lawn means making the long lawn to shorten that is realized by a “shortening” function. Similarly, Energy Product and Lawn care connect to each other via “supplying”. Visualizing these elements in the diagram makes you notice that a subject who controls Energy Product is missing. And a new stakeholder “family” appears.

The above description of the framework is converted into the following two use-cases: (i) “When an operator manoeuvre the Lawn care, the Lawn care makes the long lawn grass short by shortening the lawn grass.” and (ii) “When family controls the Lawn care, energy is supplied to the Lawn care.

The diagram could also help you find stakeholders and their relationships. In Figure 3.8, “operator” is composed of either or all of the following components: couple, parents, grandparents, kids and grandkids. As for “family”, “pets” is added to the list.

This framework also allows you to detect negative stakeholders [28] on the early stage. Figure 3.8 shows that “entire neighbourhood” was detected as a negative stakeholder and “family” could also become a negative one in certain contexts.

Use-cases for the negative stakeholders are described as follows: (iii) “operator makeovers the Lawn care, it makes noise causing damage to neighbourhood.” and (iv) “When an operator manoeuvres the Lawn care, family could get caught in the machine causing damage to family.” The system requirements should be generated keeping the traceability back to these stakeholder requirements. Meanwhile, since the above (iii) and (iv) are requirements of negative stakeholders, the system requirements generated from these use-cases should be the ones to prevent such use-cases from occurring.

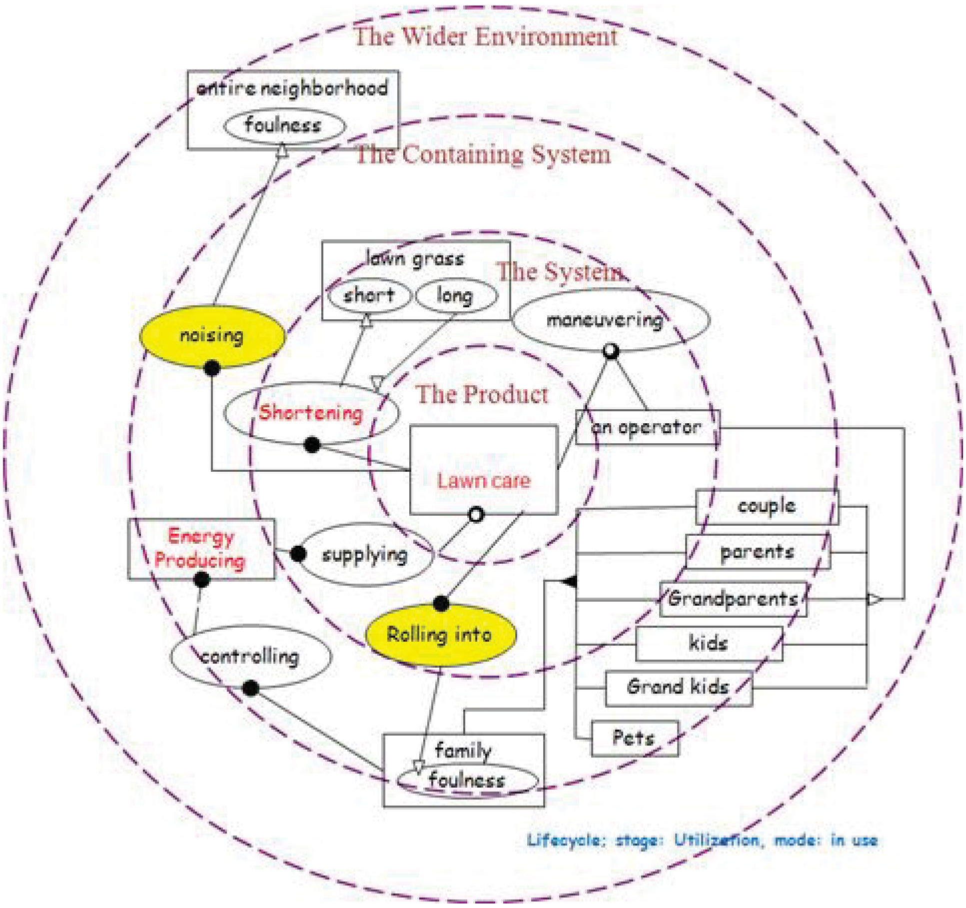


Figure 3.8 Model-Assembled Framework for Stakeholder Requirements Definition

Chapter 4

Experimental Study for the Evaluation

4.1 Actual Enterprise Information Systems Developed using Different Processes

In this study, we conducted evaluation of the developed framework. It was an academic content search engine for existing distributed campuses in the university. This system is aimed at integration of a lot of sites that could be used with the same easiness and smoothness as the centralized system [58]. The basic function should be searching for any of the information scattered on any part of the company network (in case of a company). It is considered to be an essential infrastructure system for practical use of company-wide information [59]. Such system has been built in many organizations for the last several years. So we considered it is an appropriate system for evaluation of our framework. We developed two enterprise search systems using one set of initial requirements.

In our experiment, we developed 2 enterprise search systems based on the same initial requirements. The procedure and artifacts of the Study Period are different in these two cases. In case 1, we conducted the Study Period without the framework proposed in this paper, then moved on to the establishment of systems architecture and systems design and finally to the Implementation Period. In the other case (case 2), we developed requirements using the framework in the Study Period, and continue with the following processes. In both cases the developed system was released on the Internet at the transition from the Implementation Period to the Operation Period and allowed general users to use it for 6 months.

The purpose of enterprise search systems is to provide the users with integrated different kinds of information (as same accessibility as cauterized system) [58, 59, 60]. The critical function is the search of all the information scattered across intra networks. This is thought to be an infrastructure system of the entire organization for utilization of information and KM (knowledge management). For our evaluation experiment, we adopted the establishment of an enterprise search system for academic contents within Keio University.

We developed an enterprise search system with the same goal in 2 different cases, and provided each of them to actual users. There is no difference between them except for the operations and artifacts of the Study period. If the Decision Gate, or the judging criteria for process transition, are different in those cases, our comparison result will lack credibility. Therefore we used the framework proposed by INCOSE for both cases [61]. In other words, we conducted the 4 phases of the Study Period (User Requirements, Concept Definition, System Specification and Acquisition Preparation) in order. The projects were organized as following table:

Table 4.1 Project Chater of Case 1 and Case 2 (Partial)

	Case 1	Case 2
Evaluation Period	Oct. 3 2005 – Apr. 2 2006 (6 months) Including Study period and Implementation period. Open to users from Apr. 2 2006	Oct. 5 2007 – Apr. 4 2008 (6 months) Including Study period and Implementation period. Open to users from Apr. 4 2008
Projected Budget for the Implementation Period	20 million yen in the Study Period (including hardware and software equipments, excluding labor costs).	20 million yen in the Study Period (including hardware and software equipments, excluding labor costs).
Systems Engineering Members	Systems Designer: (B.S. 5 yrs experience in an IT corporation, experience of 1 information systems development project. Skills: Java C Ruby) Systems Implementer A (Master's Program 2nd year Skills: Java, C) Systems Implementer B (Master's Program 2nd year Skills: Java, C) Systems Implementer C (Master's Program 2nd year Skills: Java, C)	Systems Designer: (M.S. 3 yrs experience in IT firm, experience of 2 information systems development project. Skills: Java C Ruby) Systems Implementer D (M.S. 3 yrs experience in an IT corporation, no experience of information systems development project. Skills: Java C Ruby) Systems Implementer E (Master's Program 2nd year Skills: Java, C) Systems Implementer F (Master's Program 1st year Skills: Java, C)
Product of the Study Period	Developed Systems Requirement, Systems Architecture and Systems Design based on Initial Requirement. Study Period Budget6% of the entire budget for the following: Period: Oct. 3 – 10 , 2005 (7 Working Days) Participant: All of Systems Engineering Members	Developed Systems Requirement, Systems Architecture and Systems Design based on Initial Requirement. Used the proposed 2x2 requirement chart during development of ConOps. Study Period Budget6% of the entire budget for the following: Period: Oct. 5 – 12 , 2007 (7 working days) Participant: All of Systems Engineering Members

4.2 Experimental Results: Periods Transition

Figure 4.1 shows the timeline of a case where the framework proposed in this paper was not employed (Case 1) and a case where it was employed (Case2). The figure shows that Case 1 finished the concept stage as scheduled while Case 2 took two more days for the same stage. In Case 1, system requirements defined by stakeholders were directly put into use without review. This is considered to be the major reason for shortening the concept stage period. In Case 2, the complicated work, after defining needs that were developed into stakeholder requirements, is also considered to be a cause for the 2-day delay in the stage.

The actual practices conducted using the matrix were (a) conducting context analysis and use-case definition, identifying a whole set of stakeholders and generating stakeholder requirements, in accordance with the grammatical rules to make sure that they could be traced back to each stakeholder and (b) developing complete building block structures (with no structural defect) in every stakeholder requirement. These are essential to complete stakeholder requirements that are verifiable and valid as mentioned in ISO/IEC 15288, which underlies ISO/IEC29148.

The quality of the concept phase could cause unexpected problems in the development phase. In [51, 62, 63], development man-hours outnumbered the schedule in both cases (Case 1 and 2). However, Case 2 successfully finished the phase in the period 4 months less than Case 1. This difference, which is to be discussed in the chapter "Discussion", could be caused by the concept phase quality.

Therefore, the framework we propose in this study could contribute to increase both the efficiency quality of the concept phase. According to [51], the cost needed for the concept phase is an average 10% of that for the development phase. In some cases of large-scale complicated projects, it jumps to some 25%. We expect that our framework could contribute to many projects.

In case 1, we allocated 6% days of Implementation Period for the Study Period. Based on the initial requirements obtained beforehand, we organized the system requirements in the Study Period. We then developed the systems architecture that achieves the requirements, and selected the optimal hardware and software COTS (Commercial on the Shelf) as components. However, the total amount was not within the initial budget. Therefore we had to add an extra 12 million yen for our implementation cost. Moreover, there were various problems during the period, which required additional 10 million yen. As a result, the total implementation cost exceeded the initial budget of 32 million yen (at the end of the period). In case 2, we also allocated 6% days of Implementation period for the Study Period. The COTS selected in case 2 was integrated prototypical search features integrated in hardware, and the total cost of implementation period was 6.8million yen below the initial budget.

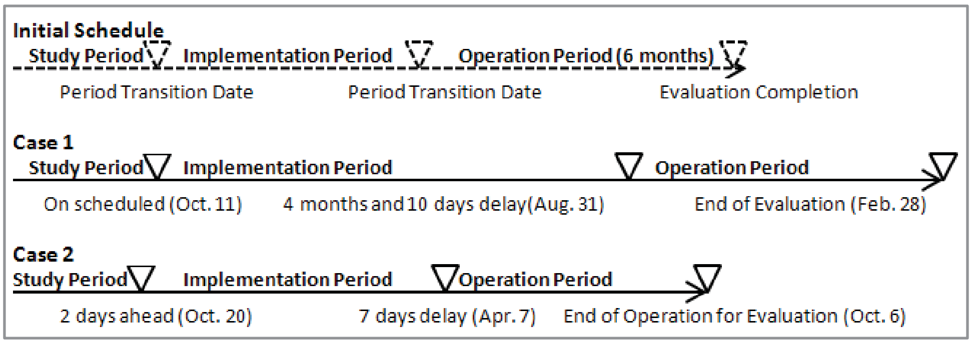


Figure 4.1 Periods Transition in the Experimental Study

4.3 Experimental Results: Identification of Requirements Specification

4.3.1 Outline

Case 1 and 2 both sorted the service systems requirements in the Concept Stage and developed systems architectures which realize the requirements. Case 2 uses the requirement development framework proposed in this paper. Table 4.2 shows some of the results from the processes. In the initial requirements, the statements “Service-Operators shall edit contents of an index database” (IR1) and “End-users shall change the order of search results” (IR2) were included. These were identified as statements belonging to the second quadrant of our chart. Then we investigated why this statement was needed and identified a statement belonging to the first quadrant. As a result, we identified statement that “End users shall get search result limited to particular publisher persons' name, particular campuses' name and particular publishing date” was the required statement. We then investigated whether IR1 and IR2 was suitable for realizing this statement. As a result, new methodologies were provided: “System shall have an interface on index database for inputting additional information, who are publishing and which campus networks are involving them, and when those information were published, on each document” and “system shall have an alternative menu for end users to get a search result limited to particular publisher persons' name, particular campuses' name and particular publishing date.” These are identified as new statements in second quadrant (Table 4.2 in bold frame). Furthermore, “System shall be maintained in 4 person hours for each functional update” was created as a statement of the fourth quadrant. Finally, hardware-software integrated search engine platform (COTS_B) was selected as a method to achieve the statements in these three quadrants.

On the other hand, in case 1, IR1 and IR2 were selected as service requirements. Each part of the search service system was to be developed as a prototype by the end users and the developers. In this case, platform software which allows modification of specifics (COTS_A_s) and platform hardware server which supports the actual operation (COTS_A_h) were selected.

Table 4.2 The Actual of Requirement Statements

<table>
<tr><th colspan="2">Requrement Development Task</th><th>The result of the initial requirment 1 (IR1)</th><th>The result of the initial requirment 2 (IR2)</th></tr>
<tr><td colspan="2">Step (1) and (2) in section 3.2: Receiving the initial requirements</td><td>IR1:
System-Operators shall edit contents of an index database.</td><td>IR2:
End-users shall change the order of search results.</td></tr>
<tr><td colspan="2">Step (3) in section 3.2: defining those statement as 2nd quardrant (ii)</td><td>Indentify IR1 as Statement in 2nd Quadrant(Function, Performance, Quality).</td><td>Indentify IR2as Statement in 2nd Quadrant(Function, Performance, Quality).</td></tr>
<tr><td rowspan="4">Results of all steps, repeatedly, in section 3.2</td><td>Defining 1st quardrant (i) statements</td><td>Produce “End users shall get information who are publishing and which campus networks are involving them, and when those information were publised, on Search Result Page”. as Statement in 1st Quadrant.</td><td>Produce “End users shall get search result limited to particular publisher persons' name, particular campuses' name and particular publishing date ”. as Statement in 1st Quadrant.</td></tr>
<tr><td>Re-defining 2nd quardrant (ii) statements</td><td>Produce “system shall have an interface on index database for inputting addtional information, who are publishing and which campus networks are involving them, and when those information were published, on each document” as Statement in 2nd Quadrant to achieve the statement mentioned above.</td><td>Produce “system shall have an alternative menu for end users to get a search result limited to particular publisher persons' name, particular campuses' name and particular publishing date ”. as Statement in 2nd Quadrant to achieve the statement mentioned above.</td></tr>
<tr><td>Defining 4th quardrant (iv) statement</td><td colspan="2">Produce “System shall be maintained in 4 person hours for each functional update” as Maintenance Statement in 4th Quadrant optimal for Statement in 1st Quadrant.</td></tr>
<tr><td>Defining 3rd quardrant (iii) statement</td><td>Select COTS_B with function enabling addition of search words to Index DB that search engine refers to.</td><td>Select COTS_B with function enabling addition of search words to Index DB that search engine refers to.</td></tr>
</table>

4.3.2 Details in Each Requirement

In both cases, 1 and 2, we conducted the two processes of ISO/IEC29148 described in 2.4, based on initial requirements. Here are some distinguishing features in the result.

Initial requirements were decomposed to provide statements. The statements included “the system shall have editing window of an index database.” (IR1) and “the system shall change the order of search result on demand of users.” (IR2) In Case 1, we adopted them as system requirement statements since they were the needs for function and performance of the system (Figure 4.2).

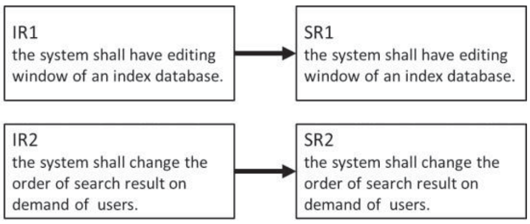


Figure 4.2 Using needs in initial requirement as system requirements: Case 1

Meanwhile in Case 2, all the requirement definition practices were conducted in accordance with the procedures provided in [64, 65]. In the 2x2 requirement chart, IR1 and IR2 were allocated in the second quadrant. We tried to figure out why these statements were needed and changed the subject of each statement to what were appropriate in the second quadrant, thus generated stakeholder requirement statements (Figure 4.3).

They were “users shall be able to search within a search result, specifying a particular author”, “users shall be able to search within a search result, specifying a particular campus” and “user shall be able to search within a search result, specifying a particular date.” We allocated the generated stakeholder requirements in building blocks and confirmed the consistency and completeness (Figure 4.4).

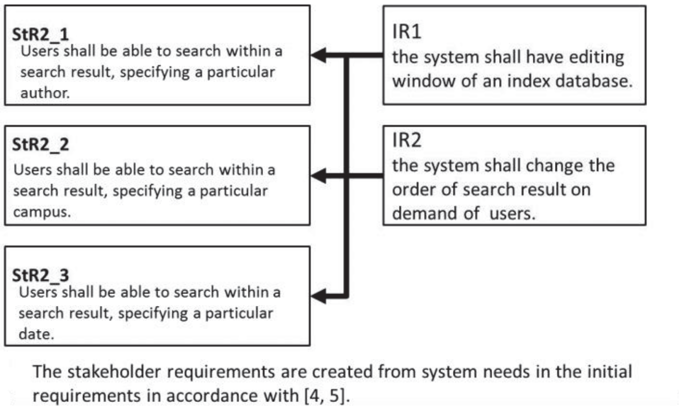


Figure 4.3 Creating Stakeholder Requirement from Needs: Case 2

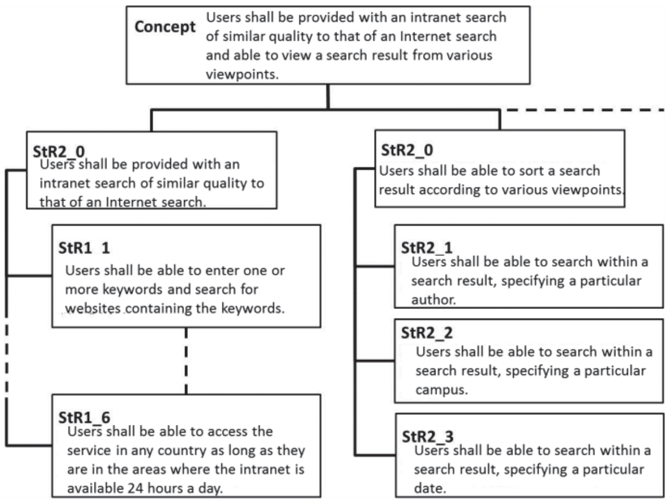


Figure 4.4 Building Blocks Made of Stakeholder Requirements (Partial): Case 2

We then studied whether IR1 and IR2 were the most appropriate implementation to realize the above statements. As a result, a new implementation was suggested (Figure 4.5). Its statements were "the system shall have interface capable of adding authors, campuses that provided information and dates when information was provided to the index database of information to be searched for." and "the system shall provide options to narrow a search result to a particular author, campus or date.” We identified those statements to be newly allocated to the second quadrant. During the discussion, another statement was generated and allocated as an operational requirement statement in the fourth quadrant. It was “Configuration update shall be done within 4 person-hours.” (the 3rd box from the top on the right column “implementation” in Figure 4.5). In Case 2 , we eventually selected a search engine infrastructure hardware-software integrated product (COTS_B). It was equipped with an enhanced interface to external components [51, 62].

Meanwhile in Case 1, IR1 and IR2 were adopted as functional requirements and two COTS’s were selected as physical components as described above. One was an infrastructure software product that allows modification to detail (COTS_A_s) and the other was an infrastructure hardware server product that secures operation in the intended use (COTS_A_h).

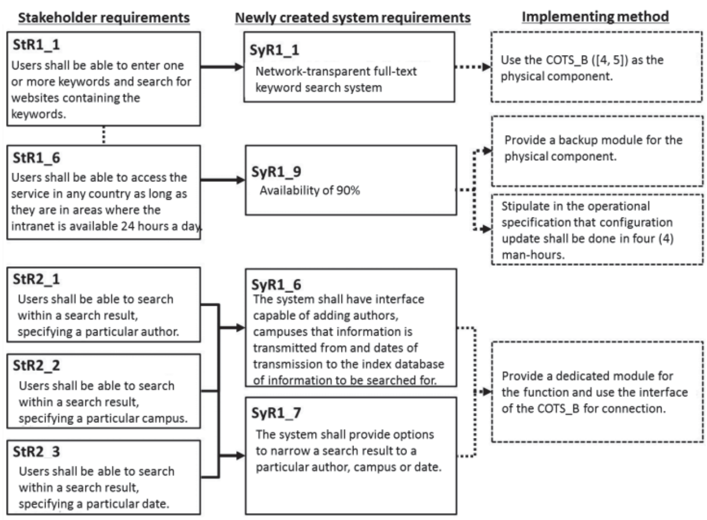


Figure 4.5 Stakeholder Requirements , System Requirements and Implementation Methods: Case 2

4.3.3 Results: Systems Architecture

The requirements' differences between case 1 and 2, caused the systems architectures' differences which were developed in the next process. Figure 4.6 shows the standard architecture of search engine. In case 1, each part was modified in the prototyping process without changing the standard architecture provided by the COTS product. On the other hand, in case 2, "End users shall get search result limited to particular publisher persons' name, particular campuses' name and particular publishing date" was identified as the ultimate goal of the service system. Therefore, (a)LDB reader identifying the publisher, campus, and date for each of the contents, (b) database for these data, (c) interface to register these data and (d) rule-base which reorganizes the search results according to the users' needs were separately developed. The statement "System shall be maintained in 4 person hours for each functional update" was also identified. We sorted the possible update processes and selected the platform software which is able to conduct all the processes with just the operation on GUI (COTS_B). Figure 4.7 is the resulting systems architecture.

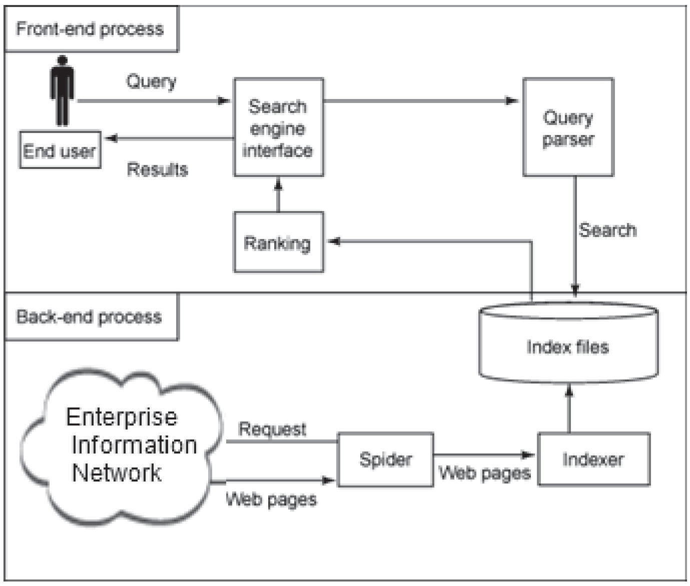


Figure 4.6 De Facto Standard Architecture of Search Engine

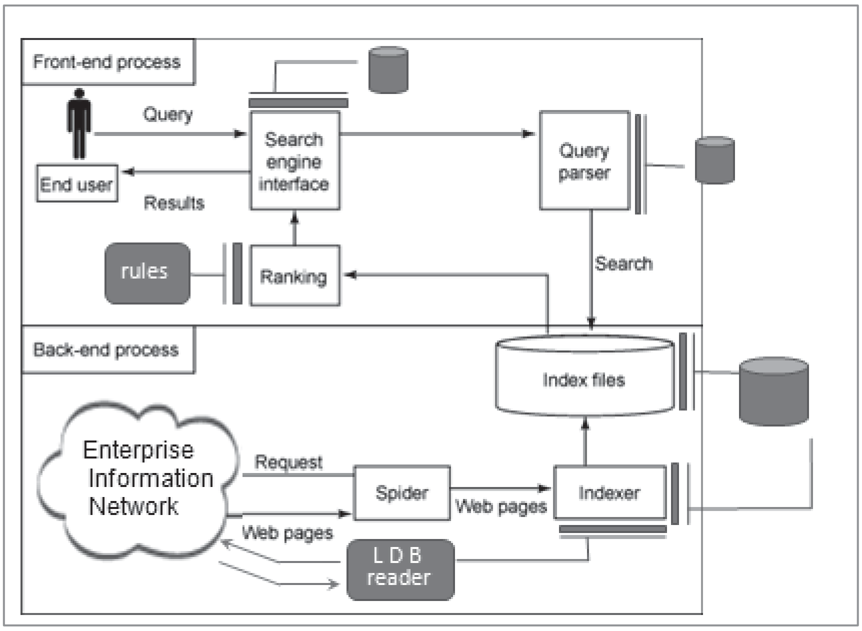


Figure 4.7 The Systems Architecture in Case 2

4.3.4 Results: Development Cost

For case 1 we selected COTS_A_s and COTS_A_h to satisfy IR1 and IR2. However, the cost of the selected components was not covered by in the budget (¥10million), therefore we had to add an extra ¥12 million for the development cost. 157 errors were found in the tests conducted during the development period. 34 errors required adjustments of the requirements, while 123 were from improper implementation. To solve these problems, we recruited 4 technicians from the provider of COTS, which required additional a ¥10 million. As a result, the total development cost exceeded the projected budget by ¥32 million (at the end of development). The COTS selected in case 2 was a hardware-software integrated type, and the price was ¥6.8 million below the expected budget. There were 38 errors in the tests conducting during the development period. 1 error required adjustment of requirements, and 37 were due to improper implementation. The repairing process was conducted by the members mentioned in Table 4.1, and this ended within the projected working hours. Figure 4.8 shows the accumulated development costs for each of the 3 steps (Fabrication, Subsystem integration, system integration).

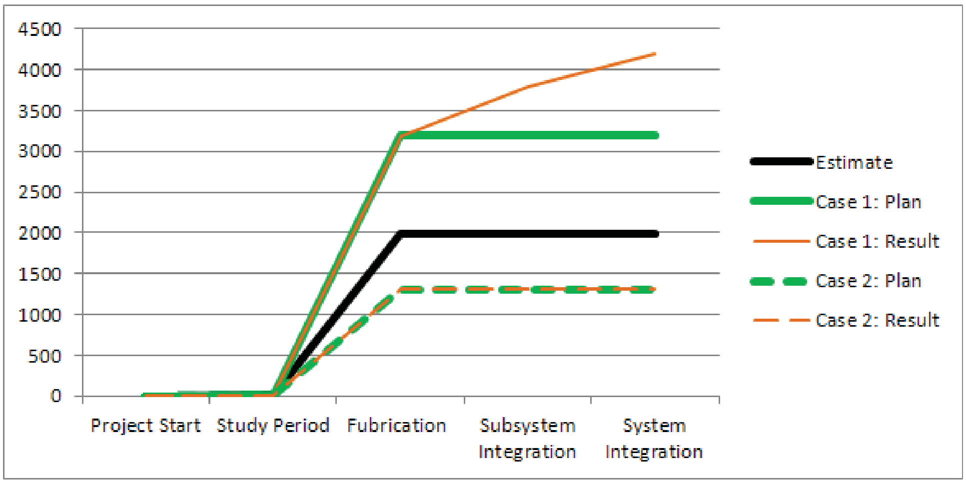


Figure 4.8 Costs of Implementation Period

4.3.5 Experimental Results: Operational Costs

We divided the operational cost as follows: (a) operational cost required to help/ support users and (b) maintenance cost required to solve unexpected problem. Table 4.3 shows working hours for each.

For the Operation Cost, “Collection of ACL (Access Control List) information” occurred once during the 6 months in both cases. The time required for this in case 2 was 1 shorter than in case 1. The task of “capturing content within RDBMS” in case 2 took 27 working hours, which is 7 times that of case 1. The average time of tasks is 7.2 worker hours in case 2. In case 1, the number is almost twice that of case 2. On the other hand, the 4 tasks excluding “capturing content within RDBMS”, took 2.3 worker hours on average, which is shorter than case 1. For the Maintenance Cost, “Rebooting entire system” occurred twice in case 1 and once in case 2. The average working hours was 4 hours in case 2. The average working hours was 18 hours in case 1. The number of case 1 is more than 4 times that of case 1. "Rebooting crawlers" occurred 102 times in case 1. In addition the working hours were approximately twice as much as in case 2. There are 3 tasks that were not performed in case 2 but were performed in case 1. These tasks took more than 20 working hours on average.

Table 4.3 All Tasks and Working Hours in the Operation Period

	Tasks	Case 1 number of occurrences	Case 1 subtotal working of hours	Case 1 time per task* (worker-hours)	Case 2 number of occurrence	Case 2 subtotal working hours	Case 2 time per task* (worker-hours)
Operations Cost	Adjustment of crawler settings	2	10.0	5.0	6	9.0	1.5
	Capturing content within RDBMS	2	8.0	4.0	1	27.0	27.0
	Capturing content within non-http site	19	38.0	2.0	1	4.0	4.0
	Capturing ACL information	1	4.0	4.0	1	3.0	3.0
	Registration of search word substitution dictionary	12	21.0	1.8	9	5.0	0.6
	Subtotal (* : average)	**36**	**81.0**	**3.4**	**18**	**48.0**	**7.2**
Maintenances Cost	Rebooting entire system	2	36.0	18.0	1	4.0	4.0
	Rebooting crawler	1	0.2	0.2	1	0.2	0.2
	Alternating index DB update timing	1	0.5	0.5	1	0.5	0.5
	Fixing system activation issues	3	62.0	20.7	0	0.0	0.0
	Fixing crawler activation issues	23	518.0	22.5	0	0.0	0.0
	Fixing contents in Index DB	0	0.0	0.0	0	0.0	0.0
	Subtotal (* : average)	**30**	**616.7**	**10.3**	**3**	**4.7**	**0.8**
Total (* : average)		**66**	**697.7**	**6.8**	**21**	**52.7**	**4.0**

ACL: Access Control List

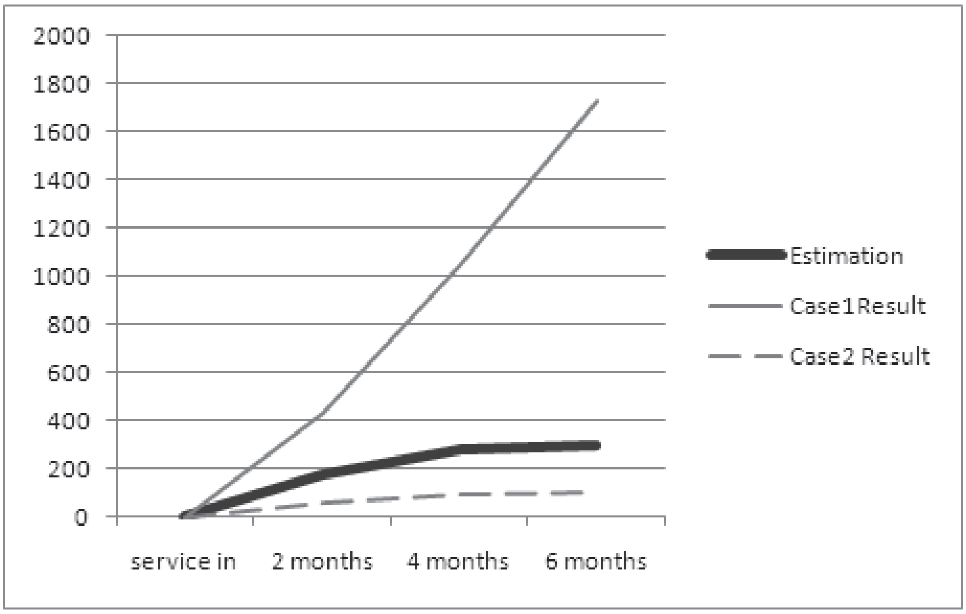


Figure 4.9 Costs of Operation Period (working hours)

4.3.6 Utilization of implemented functions

The number of functions implemented in the completed system was 18 in Case 1 and 11 in Case 2. They are the functions that users use when they directly operate the systems and do not include web browser functions such as going back to the previous page and so on. We tested the both systems for six months and found that in the Case 1 system 8 out of the 18 functions had never been used and in the Case 2 system all the functions were used.

Figure 4.10 shows how often each function was used during the 6-month test period. The graph shows the change in the number of access to each of the functions, 10 in the Case 1 system except for unused ones and 11 in the Case 2, over time. Functions in the Case 1 are shown in solid lines and those in the Case 2 in dashed lines. The explanatory notes to Figure 6 indicate the functions. The number on the left indicates the case number, 1 or 2, and that on the right indicates the function number. “1_1” means that it is the function 1 in the Case 1 system. The number of uses indicates the monthly accumulation. Only one function was implemented both in the Case 1 and 2 systems, shown as 1_1 and 2_1. It was "a function for outputting a list of all the webpages that contain (an) entered keyword(s)."

In case of the Case 1 system, users accessed to its functions around 90 times when they started using the system. The number increased in the next month but started dropping in the 3rd month. In case of the Case 2, the number of access to its functions was smaller than that in the Case 1 in the 1st month and it decreased in the 2nd month. The number started increasing in the 3rd month and almost leveled off for the last two months.

Compared to the Case 1 system, the access to each function varied widely in the Case 2 system, particularly in the latter half of the period. Meanwhile in the Case 1 system, the access to each function varied little except for one function. In the 6th month, the numbers of access to functions all fell below 20.

The total number of users during the six months was 548 for the Case 1 system and 689 for the Case 2.

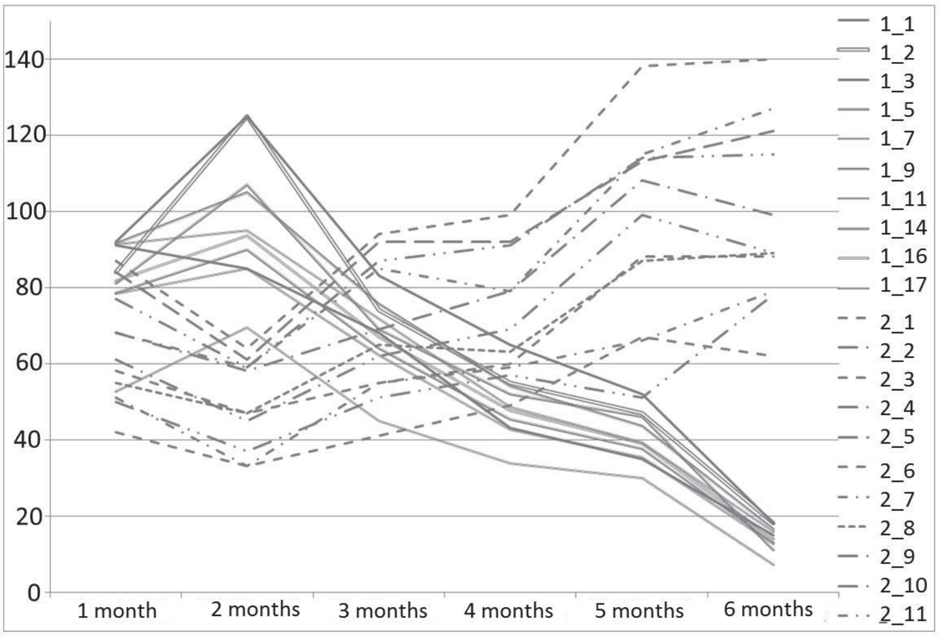


Figure 4.10 Occurrences of using functions

Chapter 5

Applying the Framework for Requirement Acquisition from Image Records of Out Patients' Activity

5.1 Background: Hospital Design

Most of the investigative research on hospital systems, especially in identifying the causes of discomfort in hospital buildings, is based on questionnaires for stakeholders. For example, a research project in Japan counted how many steps doctors took inside the hospital buildings [66]. The project reported the relation between the results and the answers from the questionnaire conducted before the investigation.

With regards to the waiting time of outpatients inside hospital buildings, the result of the questionnaire conducted by the Ministry of Health, Labor and Welfare, a Japanese government agency, and the result from questionnaire conducted by doctors are significantly different. In the former investigation, this was the major factor of dissatisfaction with hospital systems, whereas in the latter, it was concluded that this was not an important factor of hospital system evaluation [67]

Saito et al. put RFID tags on hospital staff, mainly nurses, and devices, and analysed the location data from RFID tags during the working hours [68]. They concluded that it is highly possible to make the resource management in hospitals more efficient using RFID.

Aono et al. focused on the problem of elderly patients wandering around in hospitals [69]. They put RFID tags on patients and developed a system that continuously monitors the location of patients. They also proposed an algorithm based on supervised machine learning, which detects patients that are likely to wander outside the hospital buildings.

On the other hand, Current work on relating emotions to actions and gestures captured from videos has just started. This research is divided into an area based on communication studies, which deals with what kind of emotions are perceived from certain actions, and another area based on physiology, which focuses on the what kind of effects emotional changes have on actions and gestures. The results from the two areas demonstrate a similar trend. When there is a change in actions or gestures due to emotional changes, the area of movement is small and the repetition of movement is frequent. When the magnitude of emotions becomes higher, the frequency of repetition also becomes higher.

5.2 Motivation: Needs of the New Hospital Design Methodology in order to meet the Customer demand

In the field of medicine, videos are generally used for patient observation and diagnosis support, or for eLearning. In the recent years, there are attempts to record medical practices, such as emergency treatments and surgeries, in the form of videos. There are claims, however, that these are not functioning effectively due to the lack of investigation on the methodology of recording and utilization of the videos.

In this research, we considered expanding the use of videos to hospital management. Most of the hospitals larger than the middle-scale ones in Japan are focusing on improving medical technologies and introducing newest medical devices. However, they are lagging behind in terms of efficiency and quality of hospital service compared to global levels.

We attempt to use videos to observe and analyse the current situation of hospital service in Japan.

In this experimental study, it's focused on to understand accurately the requirements and demands of outpatients, who are important stakeholders of hospital systems, towards the current systems using an original method. We follow the requirement acquisition strategy of observation described in Section 2.1. We record the actions and gestures of consumers using video cameras. This sort of an attempt has not been reported yet. With the method using RFID mentioned in Section 2.3, it is hard to identify the actual timing of the discomforting situations. We analyse the recorded videos and model the outpatients. This method avoids the problems that previous methods faced (information is collected based on predefined the user action goals), and allows the acquisition of the genuine facts. We manually conduct the analysis of videos using the results from Section 2.4. When outpatients take actions that can be identified by the method in Section 2.4, we identify what other attributes of the outpatients are affecting them.

It is also unclear whether the waiting time is affecting the discomfort of the outpatients from the results of the questionnaires (Section 2.3). We can identify this by introducing the proposed method [70].

5.3 Procedure of Recording Outpatients' traffics and motions

We recorded the movement of outpatients in a certain hospital using surveillance cameras. To keep sight of the motion paths and pauses, videos are recorded from several locations in a hospital building. We recorded the videos including early morning and late evening hours in a certain period of days, to avoid missing the differ between among the time of a day and days of a week. Therefore, we identified and traced the movement of outpatients manually by watching video data carefully. Our procedure was as follows. Some information is not opened in this paper, because of our non-disclosure agreement.

1) Period

A week including Saturday and Sunday in January to February

2) Target

A middle-scale general hospital (approximately 100 outpatients, inpatients, and staff members each), located at approximately 1 hour by public transport from one of the 5 major cities in Japan

3) Recording Method

Set up several surveillance cameras to cover the everywhere outpatients are and to record the most-to-least situations.

4) Recording time

For 1.5 hours before to an hour after the end of clinical consultation hours

5) Specifications of surveillance cameras

Colour camera with wide angle Vari-focal lens (SE-R600S from Daiwa Industries, 38megapixels, 600TV lines, horizontal angle of view 97.4°~23.8°), monitor with SD card digital recorder (DVR-MT700 from Daiwa Industries), timeline observation program stored in SD card

5.4 Preparation and observation procedure

In many domestic middle-scale hospitals, we can rarely find a situation where surveillance cameras are placed. The hospitals we were dealing with being no exception. To capture the movements of outpatients thoroughly, we hooked up multiple cameras on the ceiling temporary in the limited periods; these are mentioned in the previous section. Our preparation procedure was as follows.

1) 1 month before starting the video recording:

Obtain the layout of the target hospitals and identify the appropriate camera location.

2) 2 weeks before the starting the video recording:

Obtain an agreement to introduce the devices and personnel to the hospital, after making a presentation of the usefulness of recording to hospital managers.

3) 1 day before the starting the video recording:

Place the cameras ((5) in Section Ⅳ-A) on the ceiling. The specific drawing information may conflict with the non-disclosure agreement, and would not be revealed in this paper.

4) After the consultation hours during the video recording:

Transfer all the videos recorded on that day on the SD card to another hard disk.

5) 1 month after the video recording:

Skim through the recorded video to understand the timing of outpatient discomfort.

The following actions/motions were defined as “discomfort state”

Start a new gesture which differs from the previous motion, and repeat the movement using a part or the entire body within the range of 2 meters (e.g., knee jiggling)

6) Until today:

Prepare spreadsheet cells to record the trend of outpatients from the arrival and to the departure. Each record shows an individual outpatient. When discomfort occurred at these locations, the flag is stored in dedicated cell. All the record starts from the reception and end at the pay desk. All records were taken and input manually, by watching videos records. The information structure of the spreadsheet may conflict with the non-disclosure agreement, and the details would be omitted in this report.

5.5 Hospital Building Structure

The structure of the target hospital is as follows. However, descriptions are limited to places where outpatients can go. Furthermore, the detailed floor plan is omitted for confidentiality reasons.

- Entrance/Exit: One. All the outpatients enter and exit the hospital building using this entrance
- Consulting rooms: 5. Different departments use different rooms

- Examination room: 8. Different departments use different rooms
- Reception: 1. All the outpatients start receiving hospital service here.
- Pay desk: all the outpatients pay the bills here and leave
- Waiting room: 3 as flowers. Outpatients wait for their turn at the reception and the pay desk here
- Waiting room A: The traffic flow of outpatients is assumed to form a star shape. In other words, outpatients move out of this waiting room to elsewhere, returns, and then wait for another movement heading elsewhere.
- Waiting room B: The traffic flow of outpatients is assumed to form a star shape, too.
- Waiting room C: The flow of outpatients is assumed to form a line. In other words, outpatients move out of the waiting room to elsewhere, goes to another waiting room, and then wait for another movement heading elsewhere.

5.6 Observation Result

206 records were created as of July 27, 2012, by performing (5) and (6) in Section Ⅳ-B. In other words, we created data of the traffics and motions of 206 outpatients. Also the data contain the information whether they made any gesture signaling discomfort during the stay at the hospital.

Of all the current records, 35 have values indicating discomfort. These motions occurred in the 3 waiting rooms. The number of patients who waited in room A, B and C are 310, 38 and 186 respectively. This exceeds the total number of outpatients, 206, because patients went to these waiting rooms multiple times. The number of patients who showed signs of discomfort were 20, 3 and 12 respectively.

One patient of 35 records took discomfort motion twice in the same record, in waiting room A and B. The timing of each was 28 minutes 16 seconds and 38 minutes 24 seconds after the arrival at the waiting room, and 28 minutes 42 seconds and 1 hour 11 minutes 02 seconds after arriving at the hospital.

The average timing of taking discomfort motion of 35 records is after 28 minutes of entering any waiting room. And the average timing of taking discomfort motion of 35 records is after 56 minutes of entering in a hospital. Figure 5.1 shows how long the outpatients stayed in the waiting room before they showed signs of discomfort. The shortest case was after 1 minute 56 seconds of his/her arrival to the waiting room, and the longest one was after 1 hour 6 minutes 30 seconds. 21 showed discomfort between 10 minutes and 30 minutes after the arrival at the waiting rooms.

Figure 5.2 shows how long patients were at the hospital since their arrival (at the reception) when they showed signs of discomfort. The shortest case was 9 minutes 52 seconds after arrival. The longest case was 3 hours 13 minutes 52 seconds. 7 of 35 records are showing discomfort between 10 to 20 minutes after arrival. 5 cases showed signs of discomfort after over 1 hour and 30 minutes. The common factor of these 5 cases is that they had moved between locations such as consulting rooms, examination rooms and waiting rooms, multiple times. The numbers of transitions are 15, 8, 20, 16 and 11 respectively. All of the discomfort state was caused in one of the waiting rooms.

Additionally outpatients showed signs of discomfort were at either one of the waiting rooms for 10 to 30 minutes. Another 21 shows discomfort 40 minutes to 90 minutes after the arrival at the hospital.

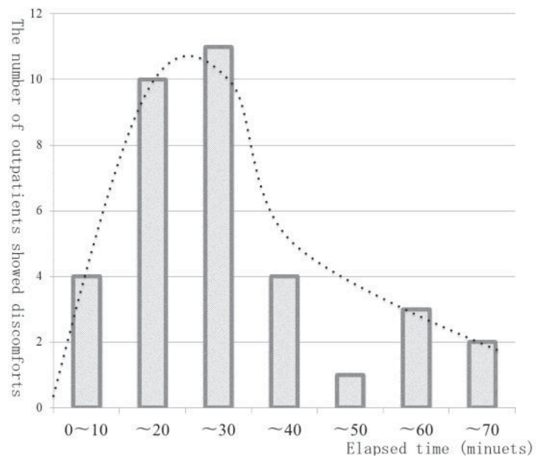


Figure 5.1 When Outpatients Show Discomfort Motions (1)

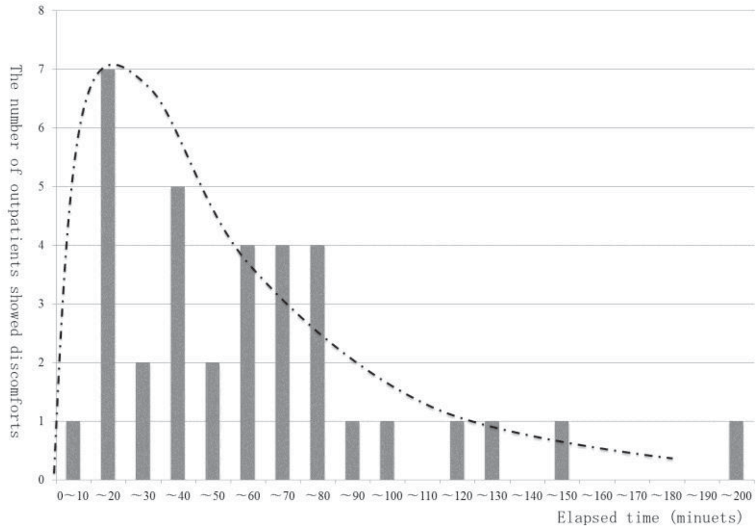


Figure 5.2 When Outpatients Show Discomfort Motions (2)

5.7 Discussion for Applying the Stakeholder Requirement Development Framework

We cannot say that the types of patients' traffic flow and discomfort motions are directly related, according to our experimental study. In waiting room A and B, which we assumed a star-shaped patient traffic flow, 23 out of 348 patients used this room showed signs of discomfort. In waiting room C where we assumed a linear patients traffic flow, 12 patients out of 186 showed signs of discomfort. This converts into 6.6% in the former case and 6.4 in the latter.

In other hand, there is a possibility that the threshold value of waiting tolerance for outpatients is around 30 minutes. This consequence was made from our observable result of the average of discomfort gestures starting timing average.

Although there was a report, in which waiting time does not have a great impact on the evaluation of hospital service (Chapter Ⅱ), we found that outpatients showed signs of discomfort from fatigue after waiting for more than the threshold time, even other attributes effects cannot be denied.

We can say that there is a possibility that long time in waiting rooms and hospitals is affecting the discomfort of outpatients. Taking into consideration the fact that hospital service is not evaluated solely by the length of waiting time, we would like to observe the relationship between waiting time and other factors.

We will especially focus on identifying the effect from the combination of multiple factors. We would also consider ways to solve issues by changing hospital design regarding these factors. We plan to combine recorded images, questionnaires and interviews so that the results are not dependent on site observation methods.

We attempted to conduct site observation using video records as a method to acquire requirements of consumers for a system. The goals of the research are to obtain contents not written in the initial requirement documents, and to remove the arbitrary factors from the bias of the questions in questionnaires. In this research we focused on hospital service to find out when the outpatients, who are important stakeholders, showed signs of discomfort. As a result, we found out that many patients were uncomfortable approximately after 30 minutes in waiting rooms. From the prior reports using questionnaires, there are those that consider waiting time as important factors, and there are those that don't. From this observation, we found out that there is a time threshold, and if the waiting time exceeds that threshold, there is a higher possibility that evaluation starts to be affected.

Chapter 6

Applying the Framework to Hands-on Education for Naive Engineers

A new hands-on type course for systems engineering was developed at Keio University, Graduate school of System Design Management. The goal of the course is to provide the students with systems engineering methods proposed by INCOSE. In this course, the framework proposed in this paper has been employed. The effeteness of the course is reported in this paper. There are two major contributions of the proposed course. The first is that we accept as many students from departments of social science as students from science and engineering. The second is that the course focuses on requirements development rather than on a certain technique or technology. These have not been realized in any of the previous engineering courses worldwide. The effects of the course on the students were analysed and it was found out that the course helped them understand systems engineering in a short amount of time. Our students tried to make systems concept, architecture and design, and procured all items for implementing their systems, in order to solve the customer's demand; they want to maneuverer an automatic cleaning machine from remote locations, even they are on their business trip.

6.1 Background

6.1.1 Motivation

We Japanese are in the midst of a national crisis. Even though there are many experts in various fields in this country, there still seems to be no clear solutions to the problems we are facing. Several years ago, Japan lost to Korea in an international bidding of a nuclear power plant in UAE. Japan is said to be lagging behind in the race for obtaining the rights to US railway construction. We believe that the reason for these situations is the lack of skills in systems engineering, not the capability of research and development within each field of expertise.

Systems engineering is defined as a technique of technology integration, which was developed in order to send men to the moon during the Apollo Program by utilizing the technology at that time. The mission of INCOSE (The International Council on Systems Engineering) is to develop the standard systems engineering methods, models and frameworks and spread them to the world. The systems engineering standards of INCOSE are based on ISO/IEC 12588. Best practice from IEEE, US Department of Defense, NASA, ECSS and various industries are also reflected in the standards. Some of the components, such as the Vee model, are deployed in Japan, but with different interpretations from those of INCOSE. Therefore the integration engineering, the essentials of systems engineering, has not yet become common. We believe that one of the reasons Japan, whose product manufacturing technologies and requirement development techniques are among the best in the world, cannot implement world-class systems is the lack of knowledge concerning systems engineering.

This chapter presents a trial course in system engineering at Graduate School of System Design and Management (SDM). This trial course provides students with hand-on education for system design using commercially-off-the-shelf (COTS) components and sizable amount of development works within the limitation in one semester (15 classes, 30slots). In the latest semester that ended February 2010, we gave an assignment to build an automatic vacuum cleaner operated by a remote site, possibly from foreign countries. Almost all the hardware was prepared by the instructors. The students were required to design the total system and to develop a part of software program based on the specifications of the COTS. Systems engineering approach was emphasized for this including mechanical engineering and information technology as well. They started with defining Concept of Operations (ConOps) and requirement analysis, and then proceeded to systems architecting and design. After the functional requirements were defined, the physical realization and feasibility were checked with verificati4on and validation planning, and detailed design was performed, and finally coding, purchasing, and manufacturing of some parts. The components actually used to be 'iRobot' and small video camera as COTS and the use of 'Wi-Fi' networking was assumed.

6.2 Outline of SDM Education

6.2.1 Objective and Basic Concept of the SDM Education

We have developed a hands-on type course that considers the actual practice to improve these situations.

To develop a hands-on type course in the master's program, we organized groups with both science and engineering students and social science students and had these groups solve the problems that we provided. This is an attempt which has not been conducted anywhere else in the world.

The objectives of the SDM are to foster strong leader for large scale projects and enterprises and creative system designer capable of planning, realizing and operating innovative systems and products, and reliable project manager creating new markets, satisfying consumer needs and operational value chain.

The basic concept of SDM is given as follows: System Design is a creative activity to bring a concept to a real being by balancing all the technical and social factors such as customer requirements, use objectives, essential functions, costs for R&D and operations, adapt to the environmental changes, and trade-offs among stakeholders. System

Management is to set up an adequate goal, to be achieved by balancing various factors including quality, cost, and schedule under risks and environmental changes. System Design and Management should be a holistic approach by observing global trends of complex interactions among diverse languages, cultures and economics.

6.2.2 Experiences and Issues

For realization of the basic concept, we have set up the curriculum including courses to teach various methodologies of systems engineering, along the lines of Vee model (Figure 6.1) It's a representative framework of systems engineering.

When one year has passed since its establishment, however, it has turned out that students with no/little experiences in industries have a difficult time in understanding and appreciating the value of system engineering with various methods and tools. This is partly because about 20% of the students are fresh from undergraduate and partly because approximately 30% of the students have very little experiences in engineering. Without such experiences, it is practically impossible to understand meanings and importance of requirement definition phase and architecture design phase that is critical to systems engineering process. This has motivated a hand-on practice.

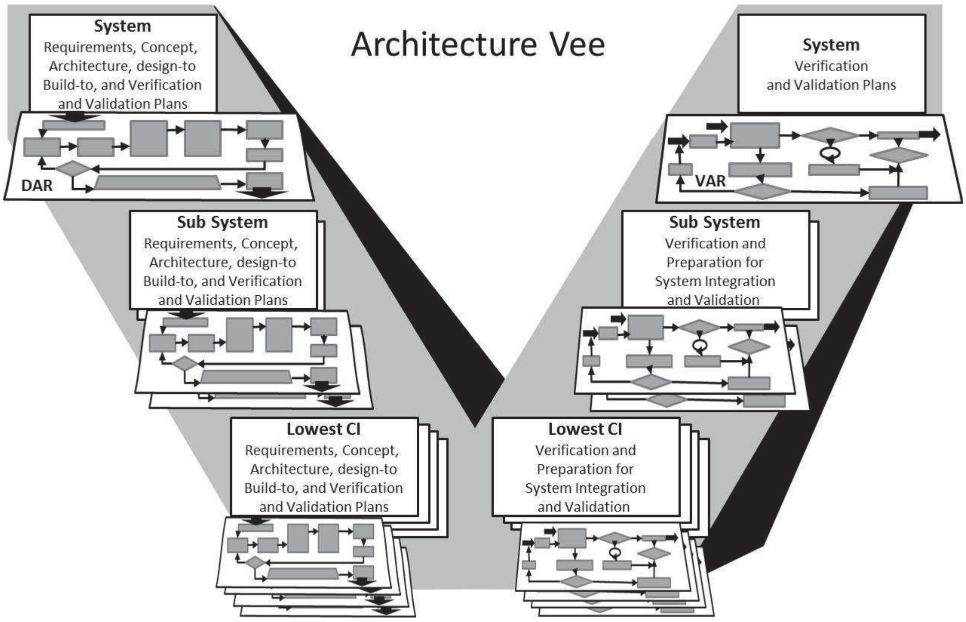


Figure 6.1 Vee Model : Modified Version of [71]

6.3 Course Overview

6.3.1 Course Schedule and Length

Student groups take an Initial requirement document, in which there is the main goal of the system. Customers want to maneuver an automatic cleaning machine, iRobot (Figure 6.2), from remote locations even they are on their business trip. Also a set of COTS provided by the instructors to each group can be combined in several ways. This class is not about reaching the only correct goal. As in the real systems engineering world, we prepare several methods to achieve the goal, and have the students go through selection criteria from multiple viewpoints. This class offers the experience of studying the really, finding the optimal solution and executing the processes of systems engineering to achieve the solution. Students should achieve an acceptance review of the supposed clients as assigned to other instructors, in the final class.

6.3.2 Course: Solution Goal

Student groups take an Initial requirement document, in which there is the main goal of the system. Customers want to maneuver an automatic cleaning machine, iRobot (Figure 6.2), from remote locations even they are on their business trip. Also a set of COTS provided by the instructors to each group can be combined in several ways. This class is not about reaching the only correct goal. As in the real systems engineering world, we prepare several methods to achieve the goal, and have the students go through selection criteria from multiple viewpoints. This class offers the experience of studying the really, finding the optimal solution and executing the processes of systems engineering to achieve the solution.

Students should achieve an acceptance review of the supposed clients as assigned to other instructors, in the final class.

6.3.3 Course: Implementation Structure

There are 3 lecturers and 1 conducts the lectures and leads the classes, 2 act as the pseudo board member and manager of sponsored company. Also there are 2 second year master students. Students receive good marks in the previous course supports. Students are in either Master's or Doctor's program at the Graduate School of System Design and Management at Keio University. 1 group consists of 4 to 6 students. Each group has students from engineering and students from social science.

Emphasis is placed not acquiring skills to develop hardware/software components based on conventional engineering techniques, but on the skills of completing a system by applying the standard approach of systems engineer under the constraints of schedule and human resources.

6.3.4 Course: Materials

"iRobot" is used, an automatic vacuum cleaner as COTS, groups are asked to implement a system in which the users are able to clean the rooms at home from abroad. This is a new product from the sponsor corporation mentioned above, and is designated as a prototype. We use the book named Visualizing Project Management [46] as the main textbook. INCOSE systems engineering handbook [6] is introduced as a reference, and if necessary, we also refer to PMBOK guide [72].

We have prepared hardware items, called the Lowest Configuration Items (LCI) beforehand in view of schedule constraint to establish a system within the 30 slots. Specifically, the instructors used COTS as LCI. For this preparation, the instructors have assumed several patterns of ConOps from the Initial requirements and devise a systems architecture which would achieve each of the ConOps.

Furthermore, the instructors prepared the LCI that would be required to establish all these system architectures before the course started. Therefore, some of the COTS that the instructors prepared may not be needed depending on the ConOps and the systems architecture that the students select.

The 3 instructors were in charge of the hands-on exercise: A professor as a customer; an associate professor as a consultant on information and communication engineering; an associate professor as a consultant in systems engineering.

Figure 6.2 Automatic Cleaning Machine ‘iRobot’ as COTS.

6.4 Our Syllabus

In the proposed course, we defined project period as Concept Stage and Development Stage in the standard system Lifecycle from ISO/IEC 15288. Following the proposal by Forsberg et al. Study Period, or Concept Stage, consists of 4 phases; User Requirement Definition Phase , Concept Definition , System Specification , Acquisition Preparation, and Implementation Period, or Development Stage consists of 3 phases; Source Selection, Development, Verification. In this course, we follow these project Lifecycle phases in order.

Class1 : Orientation

Explanation of course syllabus and overview of systems engineering. Students are asked to provide a self-evaluation on each of the systems engineering terms based on how well they can explain the term to a third person on the scale of 5. Proposals concerning the implementation method of the remotely controlled vacuum cleaning system are handed in. Students are also asked to write down the major of the undergraduate degree.

Class2 : Understanding the Importance of Requirements Development

We explain that the essence of systems engineering is to satisfy the requirements of stakeholders, and have the students practice concretizing (converting into engineering and quantitative terms) vague requirements. All the processes from this class on are conducted in groups. We explain the role of decision gate and the implementation method in this course.

Class3 : User Requirement Analysis Phase

Each group receives the identical “system concept plan” as initial requirements from the pseudo sponsor corporation and acquires the skill inventory of group members and approximate budget information. Groups are asked to improve the initial requirement documents and separate the text into those related to implementation method and those related to state change to achieve by the introduction of the system.

Class4 : User Requirement Definition Phase

The groups list up the possible stakeholders and identify the most important requirements from each stakeholder viewpoint. After the lecture on the basic structure of documents, the groups try to find inconsistency in the requirements and update the required documents with the sponsor corporation.

Class5 : User Requirement Development Phase

Groups develop ConOps and prepare multiple system implementation plans. They also conduct the decision gate to agree upon the basic requirements.

Class6 : Concept Definition Phase

Groups select few candidates for the implementation method concepts and adjust the balance between the expenses for the purchase of necessary components and the advantages of the implemented functions.

Class7 : System Specification Phase

Student groups select the implementation method concept, prepare the systems architecture and check the feasibility using models and simulations.

Class8 : System Components Specification Phase

Student groups select each of the components and implementation method of the interface. By adjusting the balance of the expenses, grapes reach an agreement with the sponsor corporation on ConOps and the decision gate.

Class9 : Acquisition Planning Phase

Student groups identify who collects/produces/assembles what at what period of time, and prepare a schedule. The groups must focus on the specifications of verification and validation and must agree with the sponsor corporation on the decision gate.

Class10 : Source Selection Phase

Student groups decide on where to buy the necessary components from and order them to fix the final amount of expense. The groups start the integration with the parts that they received.

Class11-13 : Development phase and Verification Phase

Student groups conduct system integration and verification according to the schedule.

Class14 : Preps for Final Decision Gate

Student groups conduct validation and make final adjustments to each of the documents.

Class15 : Final Decision Review (Acceptance Review)

Student groups make the final presentation and demonstration to the sponsor corporation. The sponsor corporation tests the system and groups receive the final approval.

6.5 Evaluation of our Hands-on Education

6.5.1 Examples of Developed System

We will present some characteristic output from each of the semesters.

(1) Autumn 2009

Members: a Dutch student from Delft University Technology, a French student in Master's program, a Korean-American student in Master's program, an American student in Master's program and a Japanese student in the Master's program. Their majors: management, mechanical engineering, management, financial engineering, physics (in order).

System outline :

The goal is to let the users enjoy the service daily without the knowledge of information technology or mechanisms. The web-service system is implemented based on Wi-Fi. The web camera operation is also conducted on the servers.

(2) Spring 2010

4 auditing students from the graduate school of Aerospace engineering at Nihon University and other universities.

System outline :

To utilize the skills of the members, they designed a hardware based system. By calling the number for the automatic vacuum cleaner from abroad using a cell phone, the device above the remote controller activates. The sound of the different number buttons drops corresponding bars of the device to onto the remote controller, which in turn operates the remote controller buttons.

(3) Autumn 2010

A Chinese student from Delft University of Engineering, a Japanese-American student in the Master's program, a Korean-American student in the Master's program. Their majors: electronic engineering, management, public policy (in order).

System outline :

The PC inside the room is remotely accessed using PC and PDA via the Internet. The PC in the room operates the vacuum cleaner using Bluetooth. Validation was conducted from 4 different countries.

(4) Spring 2011

4 Japanese students from Master's program. Their majors: commerce, aerospace engineering and mathematics and informatics, electronic engineering, and law. 3 of them are adult students.

System outline :

The vacuum robot and the user communicate through twitter to control the robot. Communication is conducted between PC or PDA and the PC in the room. The indoor PC sends a command from twitter to the robot using Bluetooth. They also focused on the fact that it is hard to monitor the movement of the cleaning robot through web camera when there is a time difference between the user's location and the room. By tracking the record through twitter, users can control the robot as if playing a game (Figure 6.3).

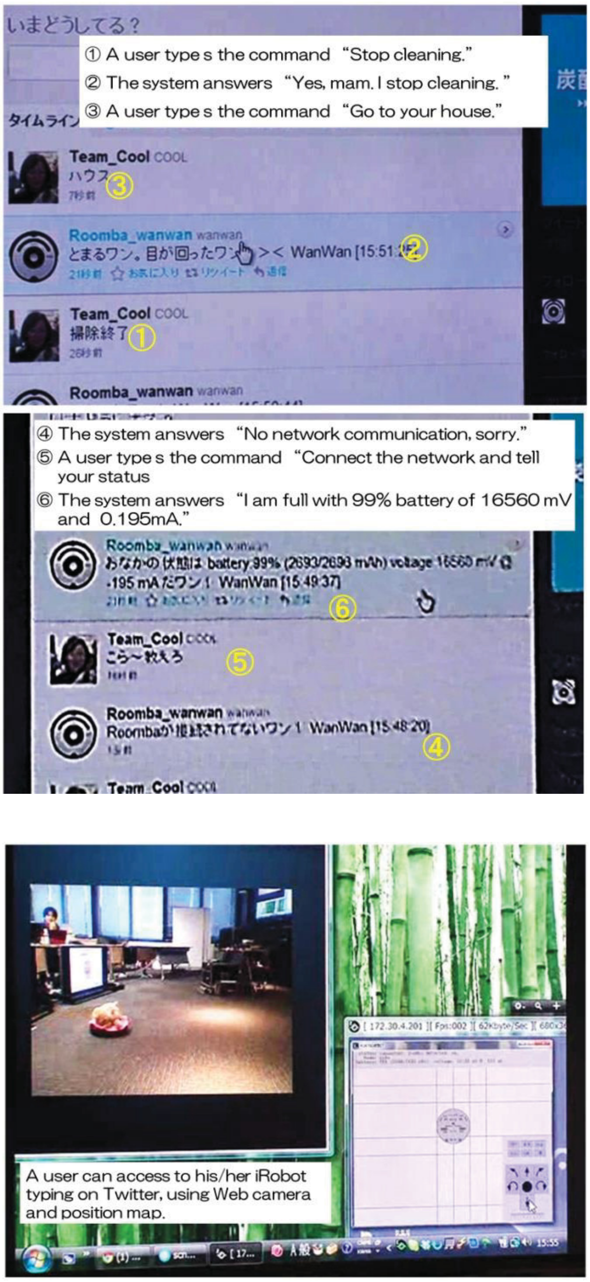


Figure 6.3 Example of Students' Systems

6.6 Observing the Students' Understanding of Systems Engineering

We asked all the students concerning their understanding of 10 important systems engineering terms, as Table 6.1 This has been conducted in the first and the final class. In the first and the last class, students recorded the understanding of each term. The understanding is on a scale of 5; 1 indicates that students know the term and 5 indicates that they can explain it to a third person. Figure 6.4 presents the results in a radar chart. The evaluation is higher in the last class in all of the semesters. We also can see that the evaluation of the initial class is higher in the autumn semester. In spring 2010, the evaluation of WBS and schedule for the final class is lower compared to other semesters.

Table 6.1 SE Term Understanding Self-Evaluation Results before and after the Hands-on Type Course

	Selected Systems Enginering Terms
1	System & Systems Engineering
2	Vee model & System Lifecycle model
3	Project Scope/ Boundary
4	Stakeholder Requirement & Systems Requirement
5	Requirement Development
6	WBS & Schedule
7	Decision Gate
8	ConOps
9	Architecture & Systems Design
10	Verification & Variation

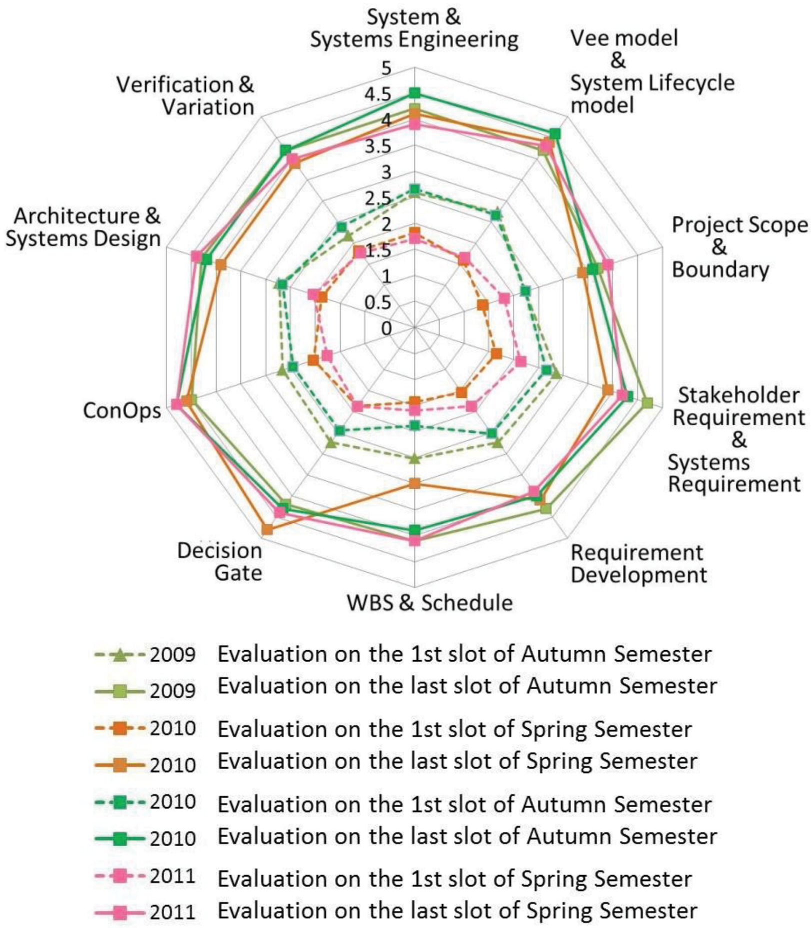


Figure 6.4 SE Term Understanding Self-Evaluation results before and after the Hands-on Type Course

6.7 Discussion and Lessons Learned

In all of the terms, we have succeeded in enhancing the knowledge students concerning systems engineering from INCOSE through the proposed course. The difference of self-evaluation in the first class in the spring semester and the autumn semester is due to the fact that the students that participate in the autumn semester are from overseas or graduates of European or American universities. In Europe and America (and even in Asia, mainly Singapore), education in INCOSE systems engineering is widely available, and the fact that students from these regions have some basic knowledge seems to have affected the evaluation results. For the spring semester of 2010, the scores of WBS and schedule seem to be lower than those of other semesters because we could not allocate enough time to these elements. In the proposed hands-on type course, we tried not to avoid homework as much as possible and have students finish the processes within a class. The lecturer adjusted the difficulty of the contents and the amount of explanation at the start of the course according to understanding of students and the progress of operations. In this semester, we allocated more time for understanding the requirements from the requests of students. We had to decrease the time spent on WBS and schedule. This, as a result, led to the difference in the evaluation.

As explained above, our hands-on course has turned out to be useful to draw potential capability of the students and also to educate them the group working and leadership. However, one of the points all of them have missed is related to the following simple question: “In what state the system is: active, sleeping or completely off?” and “How can you switch on and off?”

We have earned many lessons learned and we are planning to reflect these lessons on the next classes with the following items to be taught to the students in addition to the curriculum as explained in the previous sections.

1) Collecting information on the COTS from the suppliers and from WEB site
2) Familiarization to the hardware equipment, especially the cleaner, iRobot, and video-camera
3) Familiarization to the COTS operations by investigating the manuals in detail
4) Configuration management especially for documentation
5) Consideration of guarantees and maintenance schemes
6) ConOps of the total system

The hands-on class intentionally had students' experiential lessons learned. In other words, we do not present methods to avoid mistakes in advance. After presenting standard systems engineering processes, we let the students work with their own ideas, and when the work comes to a halt due to some problems, the instructors will then lecture the students. To be precise, we helped the students clarify the causes of the problems and gave a brief guide on what requires rework. The cause of the first rework was the lack of check on the feasibility of component integration, and the instructors assumed several integration methods and combination methods and instructed the students to select the optimal one. The second rework was due to the incompleteness of the initial requirements. Specifically, failing to limit the functional domain of remote operations ceased functioning enhancement operations on the vacuum cleaner, which is one of COTS products. The instructors told the students to sequentially conduct boundary refinement at each process of systems engineering, and students succeeded in function identification.

The lesson that instructors learned through these two rework processes is that the students who have no practical experience tend to actively propose free and intuitive implementation methods from (parts of) requirements in hands-on classes. For the ones who have practical experience have the tendency to be trapped within their experiences and avoid techniques and methods they have no experience with, there is a possibility of revolutionary breakthroughs. On the contrary, there is also a tendency that the students cannot look at other implementation methods and rush to the fabrication process without much consideration. This was another lesson learned. In real life systems engineering, we assume several implementation methods to achieve risk dispersion and process according to schedule, and select an optimal method from multiple viewpoints. Therefore, cutting-edge technologies, in many cases, are not adopted (due to insufficient feasibility). Students, who have no practical experience, tend to assume cutting edge technologies and methodologies as the only implementation method due to personal interest (in most cases of intellectual interest, which is excellent) and conduct the processes, which in turn cause rework processes.

Chapter 7

Future Works : Design Concept of Ad-hoc Information Network System for Disaster Mitigation

There is a plan to utilize the stakeholder requirement development framework proposed in this paper, to a real world system [73]. The concept of the system is introduced, in this chapter.

7.1 Background 1: Experiences of large-scale disaster on Northeast Region in Japan

Japan is a country with the advanced telecommunications infrastructure [74]. We are also equipped with the advanced medical response system at the time of disaster [75]. Despite that, Japan suffered a serious damage at the time of the Great East Japan Earthquake occurred on March 11th, 2011.

Japan's advanced telecommunications network is used for people to enjoy digital video contents and TV phone calls at normal times. At the time of large-scale disasters, it is also used as a wide-area medical emergency information system. This system is developed by the Health, Welfare and Labor Ministry and it's an application of the public communication network to emergency use. At the time of disasters, doctors and paramedics entered information about the casualties including the medical triage results into the system. Then it allocates patients to emergency hospitals across the nation to provide the appropriate medical care. At the time of 3.11 disaster, the system was there but did not work as it was intended [76].

Meanwhile, Japan's medical response system for large-scale disasters has been improved through serious disaster experiences [77]. Today, the Japan Self Defense Forces and inter-prefectural medical teams are called to form Disaster Medical Assistance Team (DMAT) at the time of large-scale disasters. At the time of 3.11, DMAT was appropriately formed and dispatched to the disaster sites in Tohoku immediately after the earthquake but failed to maximize its duty [78].

On March 11, 2011, huge tsunamis swept the vast area of Tohoku, causing a complete breakdown of all the infrastructures including telecommunications. Without telecommunications network available, the wide-area medical emergency information system failed to activate and there was no way to locate casualties who are in need of emergency medical treatment. Without such information available, it is assumed that DMAT was not able to perform its best to save lives [78].

In this paper, we propose a concept of Emergency Information Network System using the next-generation mobile telecommunications function equipped with the disaster response mode, instead of using the telecommunications infrastructure. This function enables the mobile network and terminals in ordinary use to shift into the information sharing system for a medical emergency at the time of large-scale disasters.

The disaster rescue operation is conducted in four phases: the first 72 hours, one week, one month and until the normal life is restored. At the time of large-scale disasters, the rescue operation during the first 72 hours is critical to minimize the casualties. The successful operation requires accurate information about the number of casualties, their locations and states. And such information must be acquired as soon as possible.

Our concept is to develop a network function that would connect the disaster sites and the new-generation network to communicate accurate information about the casualties during the initial rescue operation. This would significantly increase the survival rate of the casualties.

Such network function requires a constantly secured network connection. We focused on car batteries and communications function that vehicles have as one of the solutions.

7.2 Background 2: Experiences of large-scale disaster on Tohoku Region in Japan, in detail

After the tsunami struck the Tohoku region, a unit consisting of personnel from the Ground, Maritime, and Air Self-Defense force was organized following the rules for emergency situations. However, this was not enough to conduct search and rescue missions due to the fact that the main goal of the Self Defense force is to defend the country. At the same time the Self Defense force had been dispatched, there were offers of help from military forces in other countries. The actual number that had been accepted is very small. However, Japan could not designate regions that required search and rescue missions, which led to inefficient cooperation with the foreign groups during the acute phase. This is mainly due to the fact that the information available from forces other than the Self Defense force was mainly passed on by words from mouth, and was very inaccurate. Therefore, many facts were unknown unless someone went into certain regions, which forced the foreign groups to stand by. There is an emergency ICT network between the Self Defense forces, and the information that is passed through this network is accurate due to the daily training. Therefore, each member of the Self Defense force can act independently to undertake rescuing missions.

However, the commercial ICT network was devastated during the acute phase. This is due to the fact that many points were physically destroyed by the earthquake and the tsunami. The low accuracy of information affected the DMAT (Disaster Medical Assistance Team) activities, whose mission is to rescue people during the acute phase. Immediately after the tsunami, DMAT teams were dispatched from each of the prefectures and arrived within 24 hours. However, the information available was mostly passed on by words from mouth, and was very inaccurate, which led to doctors themselves running around from place to place in search of accurate information. Although there are many victims that couldn't be rescued, there were places where medical officers and vehicles from the Self Defence force were already there before the DMAT teams arrived.

The concept of soft infrastructure for disaster management is very new in the field of disaster management, civil engineering and city and town planning. Well defined architecture of concepts and standards are inevitable to reduce the cost and to facilitate the penetration of needed devices and systems and in this stage international cooperation is urgently needed before individual measures are implemented in each country and region. The objectives of the project This project deals with the technology related to disaster management based on ICT and experts' experience in case of earthquakes, volcanic eruptions, tsunamis, landslides storm and flood damages and wildfires. Aiming at enhancing the capacity of disaster management, that is developments of mitigation, preparedness, response and recovery strategies, the objective of this project is to show that introduction new technologies is needed. The management of disasters requires making hard, complex and risky decisions in very short time with a limited amount of imprecise information. Learning from past experiences can be of great help in this process if the acquired knowledge can be suitably formalized, stored and made available to who needs it at the right time in an appropriate and useful way. In the regions where disasters have recently attacked, one of the findings which was recognized as an important incorporation of lessons is that the disaster will not only impact our personal life but also gives heavy damages to the activities related to business and manufacturing. Therefore, a quick recovery implies restart of the manufacturing process under the condition of disjointed supply chain.

On the other hand, cutting edge ICT technologies are currently available, and the trend shows that they will become pervasive data sources in the near future: sensor networks, mobile telecommunication, probe vehicles and humans, image processing, data and images captured by unmanned controlled devices are just examples of these technologies.

Neither the knowledge and expertise on managing disasters nor the ICT technologies have been used so far in a well-structured formal way to assist decision makers in the management of disasters. Therefore, this project is aimed at developing the methodologies and procedures to collect in a systematic way the experts experience on managing disasters, learn from these experiences, and formalize it in terms of a Disasters Knowledge Based System. To properly manage data, appropriate data collection and fusion procedures will be designed and developed based on the techniques already developed. An efficient way of dealing with the available information requires appropriate modelling of the disaster scenario (e.g. an ad hoc transportation model scenario to deal with evacuations). ICT data, information from the Knowledge Base, and modelling analysis will be the main components of a Decision Support System (DSS) aimed at assisting disaster managers in making the best decisions and, in some cases, evaluating a priori the potential impacts of disasters.

In short, the evacuation which consists of three phases of disaster management, that is mitigation, preparedness and response will be the main subject of our study. From the point of utilizing ICT to evacuation, it can be also divided into three steps from the operational point of view; that is monitoring, decision making and transmission of or provision of necessary information to drivers and pedestrians. The last phase of disaster management is the phase of recovery. ICT can also contribute to the enhancement of recovery. However, the time constant of recovery is far more gentle than the time constraint needed for evacuation. These four phases of disaster management are supported by various engineering and academic disciplines but the integration of related academic disciplines is not well introduced in the area of disaster management. In order to design disaster management systems, an integration of various engineering technologies related to the four phases of disaster management is necessary.

The first objective of this project is to give methods to design the evacuation planning by integrating the technologies related to the three phases of disaster management. This includes the design of monitoring system using sensor technology, probe systems and integration of necessary data to GIS by processing the data and images captured by various sensor systems. The data accumulated in the GIS is transformed to the Decision Support System which consists of the formalization of the knowledge to manage disasters and the transportation model to evaluate the performance of the decisions made by decision makers for evacuation. Thirdly, the decision and related information are provided to vehicles, homes, factories, business offices, pedestrian, and the others using currently available communication devices such as smart phones, digital multimedia broadcasting systems. The second objective of this project is to propose methods to design the logistics management systems in the recovery phased. In this project, two different types of logistics management systems are considered. One type is the logistic systems for people in shelters or in isolated areas and another type is the logistics systems for recovery of supply chain management. Both systems need ICT to support inventory management and tracking and tracing of resources using RFID and mobile communication systems, for example. Therefore, one of the aims of this project is to identify the differences between disaster case operations and normal case operations in terms of logistics and inventory management, legal issues associated with the operations and discuss about the limits of commercial logistics operation models for disaster management. Therefore, the purpose of this project is to propose a new type of Decision Support System for logistics management in recovery phase. This system contains simulation tools for decision makers to test potential resource allocation and planning options. In this stage data collection during response and recovery phases can also be used in post-disaster analysis.

The third objective of this project is to design an information management system that allows users to accurately recognize the possible regions where stakeholders in need of rescue exist.

In Japan, an Emergency Medical Information system for widespread disaster had been developed under the leadership of the Ministry of Health, Labor and Welfare. It is required to master this system in order to become a DMAT member. This information system allows users to input the accurate location and conditions of those who need to be rescued using not only text information but also maps and voice data. This system also enables the user to contact the police and the fire department, and to check for vacancy in hospitals. It is also secure from external threats.

However, this system can only be operated with a broadband network. Immediately after a tsunami, the ICT network infrastructure would be severely damaged, and the system would not function at all. In Japan, we are finally able to analyse the transmission information from cars and recognize where the stakeholders in need of assistance were, two years after the earthquake. After a disaster, especially within 72 hours, we first need to set up an emergency ICT network by combining thermal sensors and satellite communication functions. Using this network, we need to identify the accurate locations requiring rescue teams, understand whether the teams other than the Self Defence force can be deployed at such locations, and make this information available to both the private teams and the Self Defence force in real time. With this system, the private teams and the Self Defence force are able to cooperate efficiently and effectively, as well as achieve large-scale rescue missions with the cooperation between East Asian nations.

7.3 Background 2: Meeting the global demand

Recent statistics tell us as that the percentage of fatalities in Asia caused by the tsunami and flood is extremely high compared to other continents (83.5%, 1980-2006). Another fact that we know is that the courses of typhoons, hurricanes and cyclones have changed their courses which were familiar in the past. Hurricane Katrina is a good example because the State of Louisiana has not experienced such a hurricane before because most of the hurricanes in the past land on the State of Florida. The case of Hurricane Sandy is another example which unexpected hurricane landed on the near Manhattan Island. These examples show that most of the cities, like New Orleans and New York were not well prepared for the coming of hurricanes. This phenomenon can be observed in many parts of Asia including Japan and this may be one of the reasons why the fatality caused by the tsunami and flood is so high in Asia.

In order to circumvent this problem many methods are proposed and known but here we will focus on the use of ICT in mitigation, preparedness, response and recovery to enhance disaster management. Now ICT such as smart-phones are easily available in Asian countries and this mobile network can be considered as a soft infrastructure to improve the disaster management of each country and region. The necessary basis for the construction of the soft infrastructure is not only to have knowledge of the technical characteristics but also to have an understanding of the process of information exchange and decision process based on the available data and information by using the soft infrastructure. The characteristic of the soft infrastructure is needed to be well understood by decision makers, various stakeholders and citizens. The need for international cooperation in Asia is very high because not only that Asia is the main victim of disasters but also it is difficult to construct the soft infrastructure in a cost efficient manner by individual country and region.

7.4 Project Concept

Our basic concept is to apply the new-generation network's flexible connection setting and operation of the emergency communication at the sites of disasters to realize constantly secured network connection.

In our system, fire engines, police cars and other local public vehicles will have a communications function equipped with the disaster response mode in order to collect information about the disaster victims and affected vehicles at the time of disasters. As candidates for communications network for this system, we focus on the conventional wireless LAN terminals and ARTB STD-T109 mobile communications system on the 700MHZ that will be available as the future safety driving assistance system. For these networks, we will develop a control function (for congestion control and information priority control) to be installed in public vehicles so that the system enables necessary information communications constantly at the chaotic and confusing situation at the disaster sites.

ARIB STD-T109 supports vehicle-to-vehicle and road-to-vehicle communications for safety driving assistance. For example, in normal use, ambulances notify that they are approaching to other vehicles using the vehicle-to-vehicle communications function and also automatically control traffic lights using the road-to-vehicle communications function in order to deliver patients quickly and safely to the hospitals. (Figure 7.1)

The new-generation network is qualified for image transmission. Using this feature, an ambulance image transmission system could be developed to communicate the patients' condition to hospitals. Also the new-generation network could improve the initial medical treatment by linking HIS (Hospital Information System) and ambulances.

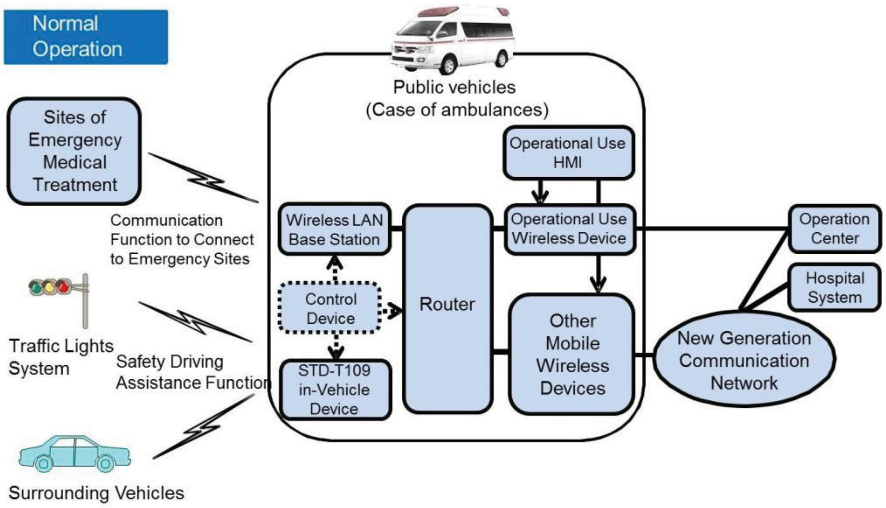


Figure 7.1 Communication Functional Diagram of the Normal Operation of Public Vehicles (Case of Ambulance)

7.5 ConOps: Concept of Operations

To add to the above normal ambulance operation, we develop the disaster response operation mode. In this mode, ambulances switch their communications function in the master station function to control communications with mobile terminals including surrounding vehicle terminals and transportable terminals.

Once the secure communication system is developed to constantly connect affected people and vehicles in disasters to local public vehicles, the information flow between the disaster response centers and those who are affected would be a flawless taking advantage of the flexible network connection settings (including the Open Flow path control) and the operation function that are built in the new-generation network. (Figure 7.2)

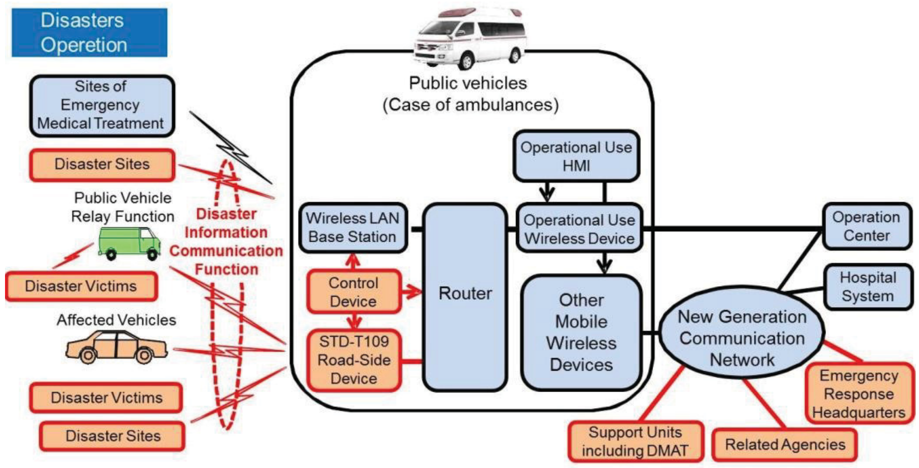


Figure 7.2 Communication Functional Diagram in Time Disasters of Public Vehicles (Case of Ambulances)

Our system's application scenarios for disaster response are:

1. Collecting data from the disaster sites via the new-generation network and making a list of those who are injured.
2. Collecting injured list from the other related agencies. If the lists have different communication and/or data formats, use XML as the common database.
3. Creating a virtual shared database to list available doctors, hospitals, medical equipment and medicines and their locations as the open data.
4. Based on the above data, the government is to appropriately allocate doctors, medical equipment and medicine via the new-generation network.

Securing the constant network connection to the terminals at the disaster sites is the key to effectively function such life-saving information sharing system at the time of large-scale disasters. Our design scope is to develop the disaster response mode for the communication system and the control function that would activate the emergency network function. This communication system must not be the one especially prepared for the disaster cases but must be the one actually in extensive use at normal times.

7.6 Development Plan

Our project will have two development phases. In the first phase (FY 2013), we will develop the communication system to collect and transmit information from the disaster sites in the following process.

i. Clarify and define required functions.

   We will first research what are required for an information sharing system at the time of large-scale disasters and then clarify the required functions of our system. These would include the disaster mode switching function, communication control function, information priority function (to give priority for information such as victims requiring rescue and care), and information identifier function.

ii. Study and identify the appropriate communication system for the disaster sites.

   We will study and identify the communication system that is appropriate in the disaster sites with the limited resources. The system must be able to connect the affected people and vehicles to the local public vehicles what will function as communication nodes. Current system candidates are the widely used wireless LAN and ARIB STD-T109 safety driving assistance network system. We will identify the appropriate system considering about the relay function.

iii. Develop the mode switching system.

   We will develop a system to switch modes (the normal mode and the disaster response mode) of the communication system installed in local public vehicles. For ARIB STD-T109 network, we will develop a system where the disaster mode would be activated when the vehicle-to-vehicle communication is shifted to the road-to-vehicle communication.

iv. Develop the communication control system for the time of disasters.

   Considering that the system would be linked to the core network in the second development phase, we will develop the control system offering the information priority control and congestion control to allow highly prioritized information to securely communicate at each access point.

v. Simulate and evaluate.

We will simulate the functions using computers and conduct evaluation.

In the second development phase (executed in 2014), we will test the connection between the communication system for the disaster sites that is developed in the first phase and the disaster response center systems for verification and validation. The tests will be conducted using the new-generation communications network test-bed, NGN-X, which is designed and operated by one of Japan's independent administrative institution, the National Institute of Information and Communications Technology.

7.7 Evaluation Plan

This project is to develop the next-generation communication function equipped with the disaster response mode. By changing the normal mode of the mobile communication network and terminals for general use into the disaster response mode, the system would be used to collect and share the disaster emergency information at the time of large-scale disasters. This information would enable rescue agencies including DMAT to conduct the rescue operation without contradiction or duplication. The system would eventually realize a significant improve in the survival rate during the initial 72-hour rescue operation.

In normal times, the mobile communication network used in the system would connect ambulances and the sites of emergency medical treatment using the wireless LAN. Doctors and paramedics would be quickly informed of the accurate emergency information at the disaster sites and be able to give appropriate medical instructions using the next-generation communication network and to stand by at medical institutions. This would also contribute to the survival rate increase. In the normal ambulance operation carrying patients to hospitals, the system would notify the ambulance approach to other vehicles and control traffic light system to make way for ambulances (This operation would require discussions with local governments.) using the vehicle-to-vehicle communication or road-to-vehicle communication function. This would assure the safety and shorten the time of operation, again contributing to the survival rate increase.

In this paper, we explain our system citing ambulances as an example of public vehicles. The system could be applied to other various public vehicles, too. When it's applied to road maintenance vehicles, they would be able to communicate between watering vehicles and cleaning vehicles enabling the echelon formation and to conduct necessary traffic light control for effective maintenance operation. The system would also provide information to private vehicles notifying about the ongoing maintenance operation. These vehicles would perform the same role as ambulances in the time of disasters.

Once our system is widely accepted and installed in various public vehicles in the future, it could be possible to apply the system to private vehicles. Taking that into account, we consider the following two evolution plans as our future system developments. One is to develop the vehicle-mounted transmission system where cars function as relay stations transmitting road information to other vehicles. The other is to develop the next negation vehicle network allowing cars to connect to multiple service centers.

These evolutionary plans would be able to provide the following effects.

1. Creating a brand new in-vehicle IT system industry (direct effect).

   In-vehicle IT system industries represented by the car navigation system industry is going through a harsh global competition. The conventional navigation system carries a dedicated stand-alone communication device and does not interconnect with any other devices. In this industry, no one is providing a solution to satisfy users' substantial needs, "We want to make the best use of available services."

   Our project is to develop a technology that could introduce a state of the art in-vehicle IT system to provide a solution to the above situation. The new in-vehicle IT system would have a versatile structure to allow any vehicle families and business industries to mount the system to create various conveniences. This business would be born in the Japanese in-vehicle IT industry and evolved into the global industry.

2. Creation of brand new service industry (secondary effect).

Having a system structure enabling vehicles to connect to multiple services would allow access of new businesses to the car industry. The conventional in-vehicle IT systems provide specified and limited services. But our evolution plan would promote access of contents providers and service providers in the car industry and various brand new service industries would be created.

Another effect of the evaluation plan could be the biggest data creation. Having IT services interconnecting vehicles and various data flowing on the network, the bigger data could be created at the site of traffic. This would promote the emergence of businesses for data analysis and application.

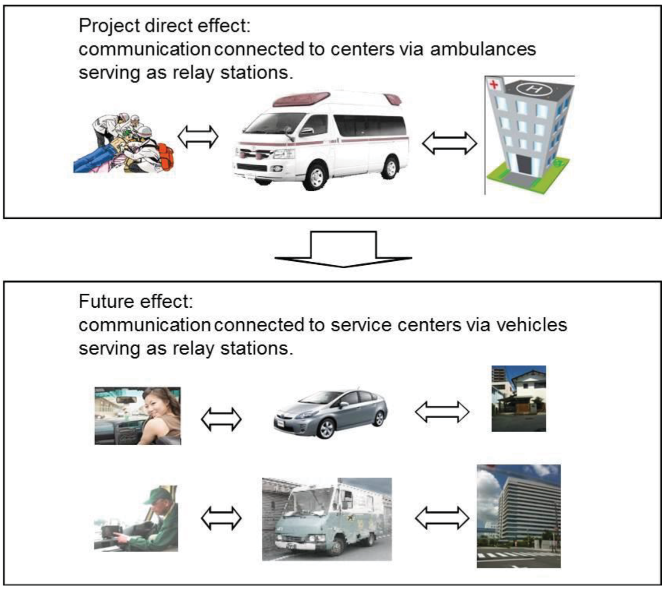


Figure 7.3 Communication Functional Diagram of the Normal Operation of Public Vehicles (Case of Ambulance)

7.8 Summary of the Future Plan

We have gone through national crises caused by earthquakes and tsunamis. Through these experiences we have established advanced communication systems and advanced disaster medical response systems. However, Japan was devastated by the Great East Japan Earthquake occurred on March 11th, 2011. Huge tsunamis swept a vast area of Tohoku causing a complete breakdown of all the infrastructures including telecommunications. Communication of emergency information was limited causing a serious delay in the initial rescue and medical operation. For the emergency rescue and medical operations, it is the most important to identify the number of casualties, their locations and states and to dispatch doctors and rescue workers from multiple organizations. In the case of the Tohoku earthquake, the dispatching mechanism and/or decision support system did not exist to allocate the appropriate number of doctors and locate disaster victims. Even though the doctors and rescue workers from multiple government organizations have their own dedicated communication system, the systems are not interoperable. In the area of the disaster management, introduction of cutting edge ICT is urgently needed. In this paper, we propose a design concept of Emergency Temporal Information Network System designed in a system of systems in the Acute Stage of Large-scale Disasters Damage Mitigation.

Chapter 8

Discussion

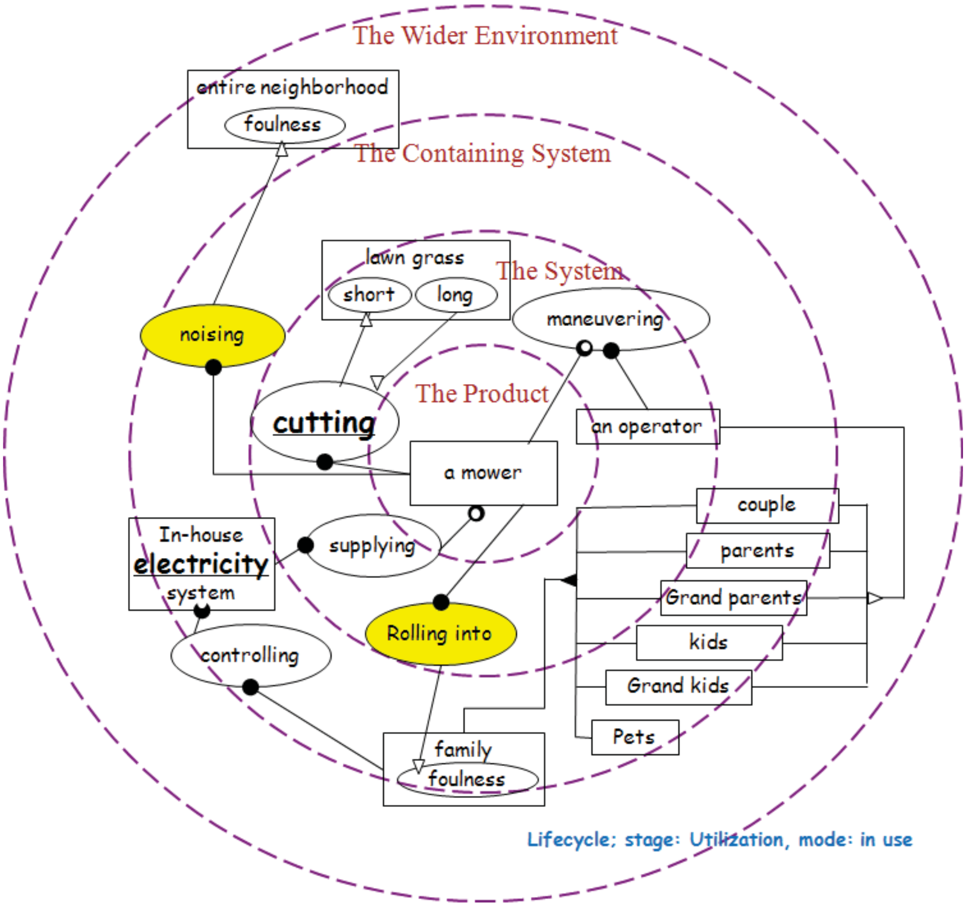
8.1 Discussion as Frameworks for Rational Requirements

The framework we developed in this study would directly contribute to the requirement definition process the for the system development which is compliant with ISO/IEC15288 and 2948. So it is considered to be effective. Meanwhile, the quality of the output could be affected depending on the experience and maturity of the system engineers. For example, distinguishing the information to be transformed to stakeholder requirements from those to system requirements is intended to make solution generation easier without depending upon technologies. Some solutions are combinations of inexpensive technologies and others are unconventional uses of legacy technologies. On the other hand, the above mentioned standards are based on Vee-model practice. Therefore the requirement process and other processes such as implementation and technology selection are conducted concurrently in the concept stage. In this situation, system engineers tend to visualize the possible implementation while proceeding with the requirement process, which is intended to be independent of implementation and technology. The result is often affected by visualized implementation and technology. This dilemma is unlikely to be solved by our framework.

An example shown in Figure 3.8 is able to be depicted as Figure. 8.1 The underlined bold words, “electricity” and “cutting” must be focused on. Suppose a mower system’s main need is to shorten the grown lawn. In this case there are several methods for acquiring power and shortening the lawn. However, we usually use electric power for a mower and cut the lawn with a blade. So these easily come up to mind and are unconsciously adopted as solutions. Once these are adopted, then technologies are to be adopted. This is not following the systems engineering basics where several concepts of implementation are given and narrowed down repeatedly at the Decision Gates.

Also, adopting the OPL result that is converted from the OPD model as a use-case would require re-transforming. For example, as mentioned in Chapter 3, such OPLs as (i) “When an operator operates the mower, the lawn’s state changes from long to short.” and (ii) “When a family member controls the energy generator, energy is supplied to the mower.” sometimes fail to be automatically generated. In those cases, transforming needs to be done by human.

By improving these two, our framework is considered to further contribute to the requirement engineering.



Example where methods for power source and shortening lawn are already specified

Figure 8.1 Another Description of Figure 3.8

8.2 Discussion about the Results of Evaluation Study

We developed a 2x2 requirement chart in order to efficiently identify the items in ConOps, a standard provided by IEEE. In this paper, we evaluated the effects of the proposed framework for the development and operational costs and utilization of the functions, by developing two identical information systems, one using this chart (case 2) and one without this chart (case 1). Based on the project evaluation from the viewpoint of QCD, case 2 was successful and case 1 failed.

8.2.1 Issue of the Cost

It's claimed that the difference of the development costs derives from the success of limiting the functions and performance necessary for the system (identification of Boundary [13]). Case 1 selected COTS that emphasizes “generality” based on ambiguous goals. Case 2 on the contrary selected COTS with interface that can combine output of external systems (not COTS that can be extended). The COTS in case 1 had a remarkable extensibility compared to the products from other vendors, but the personnel handling this product required high level of expertise. The uniqueness of the required expertise (not the level of general technique but uniqueness of products) could not be recognized beforehand, which led to unexpected expenses. The integration process for the COTS adopted in case 2 required the same level of general technique as case 1, but the product hardly required specific techniques. Furthermore, case 2 was successful at identifying the Boundary because our 2x2 chart clarified the goals of the information system (how it will be used, how it will be managed, etc.). This allowed adequate selection of methods that will be the elements of systems architecture. According to the previous research, in order to lead a project to success in terms of development cost, it was considered that 20% of development costs should be used in the Study Period. However, we succeeded with lower cost in the Study Period (6%). This is also due to the introduction of the 2x2 requirement chart, which enabled an effective development of ConOps.

In general, the accuracy of comparing actual systems is questionable even if the systems are developed in similar environments. To be specific, (a) when conducted by same personnel, the experience from previous case may be reflected, and (b) when the time period is different, the selection may change due to the advent of new products or technologies. For (a), we did not include the same member for the two cases in our experiment. For (b), we purchased the newest version of COTS used in case 1 after the first experiment, established the same system, and checked if the same problems occurred with the newest version as well. From these facts we claim that the difference of the time period did not affect the accuracy of the comparison that we conducted. There are hardly any reports where identical systems are developed using different methodology (mainly due to problems of resource). This is another contribution of our work.

8.2.1 Issue of the Rational Functions

The use of the functions widely differed between the system that employed the developed method (Case 2) and the system that didn't (Case 1) as described in Chapter 4. In Case 2, users continued to use all the functions and the number of uses increased. Based on the quantitative evaluation in the previous chapter, this chapter discusses typical comments users gave in their interviews as shown in Table 8.1 and our previous reports [51, 62]. The number of comments given by one user varied from person to person. So we grouped the comments by content. The comments were grouped into 58 in all.

At the start of the system, the Case 1 system was used much more than the Case 2 system despite the fact that the two systems targeted the same users. Most of the functions of the Case 1 system were utilized repeatedly. The Case 2 system started with low usage. The difference possibly came from a six-month time lag between the launches of the two systems [51]. Users were disappointed with the Case 1 system when the Case 2 system was launched (Table 8.1). That seemed to put off many users from using the Case 2 system. The frequency of use of the Case 1 system began to plunge three months after its launch. That was very likely to result from the problem with operation of the system described in [51].Eight functions of the Case 1 system were not utilized at all. Those functions were submenu options under the menu designed to implement IR1 and IR2. The frequency of use of the Case 2 system began to soar three months after its launch. It is very likely that the users disappointed with the first system began to appreciate the usefulness and ease of use of the second system.

The utilization of the Case 2 system slightly dipped two months after the launch. That coincided with the occurrence of the trouble during operation of the system [51]. According to this article, the trouble was the only thing that took more operational man-hours in the Case 2 system than the Case 1 system. It was likely to be temporary instability often seen shortly after introduction of a new system.

The changes in the utilizations of the functions 1_1 and 2_1 are noteworthy. Those functions are the same. They are a function for outputting a list of all the webpages that contain (an) entered keyword(s). The frequencies of utilization of both functions were almost the same at the start. But, the use of the function 1_1 increased and that of the function

2_1 decreased. The difference again seems to result from the problem with operation of the system described in [51]. The comments in Table 8.1 suggest that users doubted the reliability of the system itself and that, as a result, they didn't use a function which otherwise should have been used much more frequently.

Table 8.1　Result of interview : Top 5

	Comments	Number of people
Case 1	Simple search is placed second in the menu and it is awkward to use.	15
	The system often becomes unavailable because of the system settings.	14
	You cannot even begin a search unless you succeed in carrying out the function on the first page (metadata updating).	4
	It is unclear how to use some functions.	3
	I didn't use some of the functions I initially requested.	3
Case 2	The system is handy and useful.	28
	The first system was awkward to use and I hesitated to use the second system.	8
	It is good that the search filed is on the top page.	7
	It is handy that a search of the content can be narrowed to a particular author or campus.	6
	The system resembles Google.	6

In Case 1, the initial requirements descriptions is used that represented functional requirements as the most important system requirements (Section 4.2). The functions 1_17 and 1_16 in Figure 4.10 in Section 4.3.6 are IR1 and IR2, respectively. Their frequencies of use are low from the beginning. It is considered that extra attention had been given to those functions while the initial requirements were developed. However, the users' comments suggest that users rarely needed to use them. In Case 2, we transformed IR1 and IR2 using the 2x2 requirement chart to create ConOps information. We created system requirements that met the stakeholder requirements shown in this information. Users used all the functions and the usage rate increased. Those results illustrate a general notion that the purposes of functional requirements found in initial requirements are seldom clear and that many systems end up having functions users do not use.

In general, some functions in a system are essential even if they are low in usage. One example is the compatibility with external database systems in the Case 2 [62]. Without this function, users will not be able to search a server that does not support the HTTP protocol. This function is designed to be used by system operators not by users searching for information. It is shown as 2_8 in Figure 4.10. The number of uses is low from the beginning and it isn't raising much. For such functions, our framework does not seem to provide any support. We will work on this matter for further improvements.

8.2.3 The Essential Success Factors for Case 2

The success of Case 2 is considered to have resulted from clarification of the purposes required for an information system, for instance, how to use it and how to maintain it, by using the 2x2 requirement chart developed in this research. As the purposes were clarified, it was highly likely that we were able to choose the right means of realization that would constitute the system architecture. The result indicates that our framework was even effective in cost management. Prior research suggested that the upstream cost should be around 20% of the development cost to succeed with a project [65]. We have achieved success with much lower cost of 6% [51]. That could be another contribution of the 2x2 requirement chart. Introduction of the chart enabled efficient generation of ConOps information.

Another factor behind success was selection of physical components. In Case 2, as shown in [51, 62], the compatibility with external database systems was achieved by employing a COTS with interface fully capable of integrating outputs of external systems. The component employed in Case 1 had great scalability well beyond the range of other similar products and it required great expertise in that particular product. We did not know such expertise was necessary and we had unexpected trouble. That has possibly led to the complaint that the system often becomes unavailable because of the system settings as shown in Table 4.3.

8.2.4 Reliability of experimental evaluation

When two systems are compared, even if they are developed in the same environment, the accuracy may not be ensured. There are two factors.

(a) If the same engineers build the two systems, their experience in the first system may influence construction of the second system. (b) If the two systems are built in different periods of time, available options may differ due to advent of new technologies or new products.

In this evaluation, the two factors were eliminated that would affect the accuracy of the comparison. About the factor (a), we didn't assign the same persons to the two cases. About the factor (b), we obtained the latest version of the COTS used in Case 1 during the period of discussion after the test period. We built the system again using the latest version and confirmed that the similar problems occurred. This could mean that the time difference does not affect the accuracy of the experiment. There are very few prior studies where the same systems are built in different methods (which would cost a lot in terms of resources). In this regard, we consider our research has great significance.

Chapter 9

Conclusion

A number of cases are reported every year where ambiguous requirements are causing delays in construction and a cost increase in system development. To solve this problem, we began the study to develop a support tool to develop high quality stakeholder requirements. Requirement engineering based on ISO/IEC2914 consists of stakeholder requirement definition process and requirements analysis process. The former process generates StRS (Stakeholder Requirement Specification) document and the latter, SyRS (System Requirement Specification) document. As for the process after the a complete set of SyRS is generated, several standard tools including RTVM (Requirement Traceability and Verification Matrix) have been proposed to conduct subsequent work with ensured traceability and they have been widely in use. It was considered that increasing completeness (removing ambiguity and satisfying needs) in SyRS would require high quality StRS, since it is the input to SyRS. To realize high quality StRS, we developed a framework for systematically and seamlessly define stakeholder requirements out of stakeholder needs. The framework is a combination of conventional techniques for context analysis and use-case definition to allow for the best use of such techniques and easy adoption by their users. It's expected that introducing our framework into systems development would contribute to efficiency in high quality requirement development.

In this paper, a framework for requirement development was developed chart for efficient generation of ConOps information, a standard provided by IEEE. The proposed framework will facilitate the development of requirements without being affected by the implementation methodology or adopted technology. The framework was evaluated by establishing enterprise systems, one with and one without the framework and compared the development cost in the 2 cases.

The framework works to increase efficiency in identifying (A) the to-be situation by introduction of a system and (B) stakeholder requirements appropriately allocated in a logical structure. These are the main information in the ConOps.

Additionally, three by-products were built, through the requirement development framework research. One is the framework introducing to a systems engineering hands-on education, the others are new research projects in the real world.

References

Chapter 1.

[1] THE STANDISH GROUP, "- New Standish Group report shows more project failing and less successful projects," http://www1.standishgroup.com/newsroom/chaos_2009.php, Apr. 23, 2009 [Jan. 31, 2012].

[2] Mizoguchi. S. (2004, Mar.) Cost Management of Information Systems in Japanese Companies: Theory and Practices. Asian In Extensor. [online]. pp.1-13. Available: http://www.iae.univ-poitiers.fr/EURO-ASIE/Docs/Asia-in-Extenso-Mizoguchi-March2004.pdf

[3] Twanson. E. B. and Beath. C. M, MAINTAINING INFORMATION SYSTEMS IN ORGANIZATIONS, John Wiley & Son, Chichester, UK: 1989, pp.4-9

[4] Glass. R. L. (2002, Nov.) Predicting future maintenance cost, and how we're doing it wrong. Software, IEEE, 19(6), pp.112-113.

[5] Talby. D, Hazzan. O, Dubinsky. Y and Keren. A. (2006, Aug.) Agile software testing in a large-scale project. Journal of IEEE software, 23(4), pp. 30-37.

[6] Systems Engineering Handbook, A guide for system life cycle processes and activities: INCOSE-TP-2003-002-3.2, International Council Systems Engineering (INCOSE), San Diego, CA, 2011.

[7] Forsberg. K, Mooz. H and Cottenerman. H, Visualizing Project Management: Charts and Frameworks for Mastering Complex systems, 3rd ed. Hoboken, NJ: Wiley, 2005, pp. 84-89

[8] Halligan. R. Project Performance International. A course presented over five days, Topic: "OCD&CONOPS in Capability Development" Old Adelaide Inn, Adelaide, Australia, August. 9-13, 2010.

[9] Roger. S, Software Engineering, 6^{th} ed. New York, NY: McGraw-Hill Companies, 2005.

[10] Systems engineering － Requirement engineering process, IEEE Standard 29148, 2011. pp.6-9

[11] Tamai. T, "Process of Software Evolution (Invited Paper)," presented at the Proc. The First International Symposium on Cyber Worlds (CW2002), Tokyo, Japan, Nov. 8-15, 2002.

Chapter 2.

[12] Regnell. B, Kamsties. E and Gevasi. V, "Summary of the 10th Anniversary Workshop on Requirement Engineering," Foundation for Software Quality (REFSQ'2004), Riga, Latvia, 2004, June, pp. 7.

[13] Suchman. L, Plans and Situated Actions: The Problem of Human-Machine Communication, New York, NY: Cambridge University Press, 1987.

[14] Regnell, B., Kamsties, E. and Gevasi, V.: Summary of the 10th Anniversary Workshop on Requirement Engineering: Foundation for Software Quality (REFSQ'2004), 2004, June, pp.7.

[15] Zave. P. (1997, Dec.) Classification of Research Efforts in Requirements Engineering. ACM Computing Surveys. 29(4), pp.315-321.

[16] Yu. E. S and Mylopoulos. J. P, "An actor dependency model of organizational work: with application to business process reengineering" in Proc. Conference on Organizational Computing System (COOCS 93), Hayward, CA, Nov. 1993, pp. 258-268.

[17] Chung. L and Sampaio do Prado Leite. J. C, "Conceptual Modeling: Foundations and Applications Lecture Notes in Computer Science. 5600" in Non-Functional Requirements in Software Engineering, New York, NY: Springer,2009, pp. 363-379.

[18] Yamamoto. S. (2008, Apr.) Requirements Engineering: NFR and Goal Oriented Requirements Definition (In Japanese). IPSJ Magazine. 49 (4), pp. 371-379.

[19] Roland,C. and Salinesi, C, "Modeling Goals and Reasoning with Them" in Engineering and Managing Software Requirements, New York, NY: Springer, 2005, pp. 189-217.

[20] Pressman. R. S, NISHI. Y, et al (Translation Supervisor) and Furusawa. S, et al (translator), Jissen Software engineering Software professional no tame no kihon chishiki (Software engineering : a practitioner's approach), 6th ed., Tokyo, Japan: Union of Japanese Scientists and Engineers, 2009 (In Japanese).

[21] Systems engineering − Application and, management of the systems engineering process, IEEE Standard 1220, 2005.

[22] Forsberg. K, Mooz. H and Cottenerman. H, Visualizing Project Management: Charts and Frameworks for Mastering Complex systems, 3rd ed. Hoboken, NJ: Wiley, 2005, pp. 375

[23] Forsberg. K, Mooz. H and Cottenerman. H, Visualizing Project Management: Charts and Frameworks for Mastering Complex systems, 3rd ed. Hoboken, NJ: Wiley, 2005, pp. 161

[24] Systems Engineering Handbook, A guide for system life cycle processes and activities: INCOSE-TP-2003-002-3.2, International Council Systems Engineering (INCOSE), San Diego, CA, 2011 pp. 93.
[25] Systems engineering − Requirement engineering process, IEEE Standard 29148, 2011. pp.6-9
[26] Systems and software engineering - Vocabulary, IEEE Standard 24765, 2010.
[27] Systems and software engineering - Architecture description, IEEE Standard 42010, 2011.
[28] Alexander. I. (2005, Month) A Taxonomy of Stakeholders: Human Roles in System Development. International Journal of Technology and Human Interaction, 1(1), pp.23-59
[29] Kevin. B. et al (Eds.), A Guide to the Business Analysis Body of Knowledge (BABOK Guide) Version 2.0. Toronto, Canada: International Institute of Business Analysis, 2010, pp.29-30
[30] IEEE Standard for Functional Modeling Language-Syntax and Semantics for IDEF0, (1998) IEEE Standard 1320.1-1998(R2004), 1998.
[31] Chen. P. (1976, Oct.) The Entity-Relationship Model – Toward a Unified View of Data. ACM Transactions on Database Systems 1(1), pp. 9-36
[32] Parsaye. K, Chignell. M, Khoshafian. S, Won. H and Chingnell. M, Intelligent Databases: Object Oriented, Deductive Hypermedia Technologies, Hoboken, NJ: John Wiley & Sons, 1989.
[33] Sakai. H, "Entity-Relationship Approach to the Conceptual Schema Design," in Proc. 1980 ACM SIGMOD International Conference on Management of Data, Santa Monica, CA, 1980, pp. 1-8.
[34] Scheuermann. P, Schiffner. G and Weber. H, "Abstraction Capabilities and Invariant Properties Modeling within the Entity-Relationship Approach," in Proc. Int. Conf. on Entity-Relationship Approach to Systems Analysis and Design, Santa Monica, CA, 1979, pp. 121-140.
[35] Sakai. H, "A Unified Approach to the Logical Design of a Hierarchical Data Model," in Proc. Int. Conf. on Entity-Relationship Approach to Systems Analysis and Design, Santa Monica, CA, 1979, pp. 61-74.
[36] Sakai. H, "On the Optimization of the Entity-Relationship Model," in Proc. 3rd USA–JAPAN Computer Conference, San Francisco, CA, 1978, pp. 145-149.
[37] The Institute of Electrical and Electronics Engineers, Inc. (2004) IEEE Std.15288 -2011: Systems Engineering – System Life Cycle Processes, New York, IEEE.

[38] Dori. D. (1995, Month) Object-Process Analysis: Maintaining the Balance between System Structure and Behavio. Journal of Logic and Computation, 5(2), pp.227-249

[39] Reinhartz-Berger, I. and Dori, D. "Object-Process Methodology (OPM) vs. UML: A Code Generation Perspective" in Proc. CAiSE'04 Workshops, Riga, Latvia, 2004, pp.275-286.

[40] Froyd. J. E, Wankat. P.C and Smith. K. A. (2012, May.) Five Major Shifts in 100 Years of Engineering Education, Proceedings of IEEE, 100, pp. 1361-1375.

[41] G. M. Bonnema, I. F. Lutters-Weustink and F. J. A. M van Houten, "Introducing systems engineering to industrial design engineering students with hands-on experience," 18th Int.l Conf. Systems Engineering: Icseng 2005, Las Vegas, 2005, pp.408-413.

[42] E. A. Gonzalez, M. C. G. Leonor, P. A. T. Mangulabnan, J. J. S. L. C. Kau and M.W.U. Reyes, "Work in progress _ an educational tool for teaching linear and control systems," 2007. FIE '07. 37th Annual Frontiers In Education Conference - Global Engineering: Knowledge Without Borders, Opportunities Without Passports, Milwaukee, WI , 2007.

[43] Castles. R, Lohani. V. K and Kachroo. P, "Utilizing hands-on learning to facilitate progression through Bloom's taxonomy within the first semester," 2009.FIE '09. 39th IEEE Frontiers in Education Conference, San Antonio, Texas, 2009, pp. 1-5.

[44] Yanfei. Liu and Pomalaza-Raez. C, "Concept learning embedded in a freshman engineering project on energy scaEducation Journal 2013, 2(3): 64-71 71 venging," 2010 Int. Conf. E-Health Networking, pp. 360-363, Shenzhen, China, 2010.

[45] Schilling. K. (2006, Jul.) Design of pico-satellites for education in systems engineering. IEEE Aerospace and Electronic Systems Magazine, 21, pp. S_9- S_14.

[46] Forsberg. K, Mooz. H and Cottenerman. H, Visualizing Project Management: Charts and Frameworks for Mastering Complex systems, 3rd ed. Hoboken, NJ: Wiley, 2005.

[47] Forsberg. K, Mooz. H and Cottenerman. H, Visualizing Project Management: Charts and Frameworks for Mastering Complex systems, 3rd ed. Hoboken, NJ: Wiley, 2005, pp. 16-17

[48] Forsberg. K, Mooz. H and Cottenerman. H, Visualizing Project Management: Charts and Frameworks for Mastering Complex systems, 3rd ed. Hoboken, NJ: Wiley, 2005, pp. 26-27

[49] Chung, L., Sampaio do Prado Leite, J.C. (2009) 'On Non-Functional Requirements in Software Engineering', Conceptual Modeling: Foundations and Applications Lecture Notes in Computer Science, Vol.5600, pp.363-379

[50] Forsberg. K, Mooz. H and Cottenerman. H, Visualizing Project Management: Charts and Frameworks for Mastering Complex systems, 3rd ed. Hoboken, NJ: Wiley, 2005, pp. 240-250

Chapter 3.

[51] Shimazu. K, Takano. K and Furukawa. K. (2011) Proposal of a 2 × 2 requirement Chart for Supporting ConOps Development to Reduce both Planning Time and Operation Cost. Journal of Information Processing, 52(2), pp. 670-679. (in Japanese)

[52] Forsberg. K, Mooz. H and Cottenerman. H, Visualizing Project Management: Charts and Frameworks for Mastering Complex systems, 3rd ed. Hoboken, NJ: Wiley, 2005, pp. 3-7

[53] Halligan. R. Project Performance International. A course presented over five days, Topic: "OCD&CONOPS in Capability Development" Old Adelaide Inn, Adelaide, Australia, August. 9-13, 2010.

[54] Systems and software engineering — Architecture description, IEEE Standard 42010 -2011, 2011.

[55] Systems and software engineering — Architecture description, IEEE Standard 42010-2011, 2011, pp. 5.

[56] R. Monson-Haefel, 97 Things Every Software Architect Should Know, Sebastopol, CA: O'Reilly, 2009

[57] Shimazu. K, Momma. A and Furukawa. K. (2003) DAISY, an RER Model Based Interface for RDB to ILP. Proc. of the 22nd International Conference on Conceptual Modeling Program Committee. Lecture Notes in Computer Science, pp. 390-404.

Chapter 4.

[58] Lewis. B. (2007) Guest Editor's Introduction: A Glimpse at the Future of Enterprise Search. IT Professional, 9(1), pp.12-13.

[59] Chun-Che. H and Chia-Ming. K. (2003) The transformation and search of semi-structured knowledge in organizations. Journal of Knowledge Management, 7(4), pp. 106-123.

[60] Hawking. D. (2006, June) Web search Engin: Part 1. ACM press. [online]. pp.1-13. Available: http://web.mst.edu/~ercal/253/Papers/WebSearchEngines-1.pdf

[61] Forsberg. K, Mooz. H and Cottenerman. H, Visualizing Project Management: Charts and Frameworks for Mastering Complex systems, 3rd ed. Hoboken, NJ: Wiley, 2005, pp. 99-102
[62] Shimazu. K and Furukawa. K. (2012) The 2x2 Requirement Chart for ConOps Development to Optimize both Cost of Study and Operation Periods (In Japanese). Journal of Digital Practices, 3(2), pp. 164-172.
[63] Shimazu. K. (2013) Contribution of 2x2 Requirement-Chart to Successful Software Systems (in Japanese). The Journal of the Institute of Electronics, Information and Communication Engineers (in printing).
[64] Systems Engineering Handbook, A guide for system life cycle processes and activities: INCOSE-TP-2003-002-3.2, International Council Systems Engineering (INCOSE), San Diego, CA, 2011 pp. 7.
[65] Forsberg. K, Mooz. H and Cottenerman. H, Visualizing Project Management: Charts and Frameworks for Mastering Complex systems, 3rd ed. Hoboken, NJ: Wiley, 2005, pp. 84-89

Chapter 5.

[66] Suchman. L, Plans and Situated Actions: The Problem of Human-Machine Communication, New York, NY: Cambridge University Press, 1987.
[67] Schuler. D and Namioka. A (Eds.), Participatory Design: Principles and Practices, Hillsdale, NJ: Lawrence Erlbaum Associates, 1993.
[68] Regnell. B, Kamsties. E and Gevasi. V, "Summary of the 10th Anniversary Workshop on Requirement Engineering" Foundation for Software Quality (REFSQ' 2004), Riga, Latvia, 2004, pp. 7.
[69] Wen Chung-Lin., Chen Bing-Yu and Sato. Y. (2010, April) Video Segmentation with Motion Smoothness, IEICE Transactions on Information and Systems, E93-D(4), pp. 873-881.
[70] Shimazu. K, Ikeda. K, Ito. M and Onoe. A, "Experimental Study of Video Records Utilization for Hospital Service Management -When Do Outpatients Make Discomfort Gestures?-," Proc. 5th Int. Congress on Image and Signal Processing (CISP 2012), Chongqing, China, 2012.

Chapter 6.

[71] Forsberg. K, Mooz. H and Cottenerman. H, Visualizing Project Management: Charts and Frameworks for Mastering Complex systems, 3rd ed. Hoboken, NJ: Wiley, 2005, pp. 148

[72] Project Management Institute, A GUIDE TO THE PROJECT MANAGEMENT BODY OF KNOWLEDGE (PMBOK GUIDE), 4th ed., Atlanta, GA: Project Management Institute Inc., 2008.

Chapter 7.

[73] Shimazu, K, “Project Concept: Design Concept of Ad-hoc Information Network System for Disaster Mitigation,” Proc. International Workshop on Vehicular Communication Systems and Networks (VCSN 2013), Mayer, India, 2013. (in printing)

[74] Yang. Daqing. (2013, Feburary) Technology of Empire: Telecommunications and Japanese Expansion in Asia, 1883–1945. The Journal of Asian Studies, 72(1), pp. 206-208.

[75] Nakajima. K, Kurata. Y and Takeda. H. (2005) A web-based incident reporting system and multidisciplinary collaborative projects for patient safety in a Japanese hospital. Qual Saf Health Care, 14, pp. 123-129.

[76] Inokuchi. R, Sato. H, Nakajima. S, Shinohara. K, Nakamura. K, Gunshin. M, Hiruma. T, Ishii. T, Matsubara. T, Kitsuta. Y and Yahagi. N. (2013) Development of information systems and clinical decision support systems for emergency departments: a long road ahead for Japan. Emergency Medicine Journal, 30(5).

[77] Suzuki. I and Kaneko. Y. (2013) Government Institutions Available at Time of the 3.11 Disaster for the Emergency Management: in Japan’s Disaster Governance: Series of Public Administration. Governance and Globalization, 4, pp. 103-106.

[78] Suzuki. I and Kaneko. Y. (2013) Government Institutions Available at Time of the 3.11 Disaster for the Emergency Management: in Japan’s Disaster Governance: Series of Public Administration. Governance and Globalization, 4, pp. 25-38.