Improving Use Case Definition using Enterprise Communication Networks

Juan M. Luzuriaga

Universidad Nacional del Comahue,
Departamento de Ciencias de la Computación,
Buenos Aires 1400, 8300 Neuquén, Argentina
jluzuria@uncoma.edu.ar

Abstract*. Although researchers noted the importance of effective communication among stakeholders, it continues to be a challenge for requirements engineering. Communication can be considered as a management tool since communication allows organisation's personnel to produce a cohesive enterprise view. In this paper, we briefly introduce our proposal on using enterprise communication networks to facilitate domain understanding. Then, we illustrate how to take advantage of these networks to improve use case definition.*

Keywords: Requirements Engineering - Requirements Elicitation - Stakeholder Communication - Use Cases

1 Introduction

Elicitation is exploratory phase within the requirements engineering process. It typically includes an examination of the organisation into which the target system will be placed; an elaboration of the system's goals; a determination of any constraints on the project; and a determination of the existence of similar systems. As an exploratory phase, eliciting requirements involves perceiving reality in such a way that people's behaviour is examined to explain how pieces, working all together, make a system.

During elicitation, a system analyst has a dual task: he or she must be a communicator who should understand and respond to what is found in observing and talking to those who are commissioning a new system or who will be the end-users of it. Above all, a system analyst must be able to perceive correctly what is needed; but organisational domains into which such software is introduced are often too intricate to be fully understood. In this context, elicitation is generally reduced to be a simple matter of interviewing or analysing documents; however several other elicitation methods are available [1].

It is widely recognised that communication problems are a major issue of software projects [2,3]. Moreover, communication can be considered as a management tool since communication allows organisation's personnel to produce a cohesive enterprise view. Communication facilitates commitment by avoiding defining confronting goals, and it also contributes to make organisation's processes more flexible. Communication is present everywhere and it also constitutes a source of power. However, although researchers noted the importance of effective communication among stakeholders, it continues being a challenge to requirements engineering.

To address this challenge, we have defined a set of heuristics and strategies for improving the requirements elicitation process. Principles, such as *transparent communication* [4], have been used to detect how to gather information and how to define requirements in such a way that feedback and user's support are always encouraged. We have specially focused on communication features that make individual works contribute to organisation's goals. Our proposal is based on certain enterprise communication networks that allow information flows. In spite of different types of organizations exist, it is possible to found *hierarchical, information and expert* communication networks [5,6].

From communication networks and from ideas and tools commonly used to manage quality systems [7], we have defined a strategy to elicit requirements and we have applied it to different domains such as health care systems, and manufacturing systems [8]. The application of the strategy produced clearer specifications but more importantly, specifications where all stakeholders committed requirements. Our strategy can be used along with many other techniques; for example, the contextual-inquiry technique [9]. Two kinds of products are generated when our strategy is applied: (1) a diagram of each enterprise network written in UML notation for deployment diagrams[10], and (2) bi-dimensional tables representing the communication flows.

In this paper, we extend our strategy to show how communication networks might help improve use case definition. Section 2 briefly introduces the strategy. Then, section 3 addresses the use case definition extension. Discussion is provided in section 4. Finally, conclusion and future work are addressed.

2 A Communication-Based Strategy to Requirements Elicitation

Enterprise communication networks constitute the formal and informal structure on which enterprise communication flows. An enterprise has its owns formal communication channels that are called *communicating flows* or *vectors* used to transmit messages throughout the organisation [4].

In spite of different types of organisations exist, the following enterprise communication networks always can be found:

- *Hierarchical Network*: a formal organisation that is represented as a hierarchy connecting managers to subordinates – or a diagram known as flow-chart. In general, this network is not effective enough to allow information flows.
- *Information Network*: information related to organisation's functions. This view describes relationships among the information used by the enterprise. It includes all information forms and notes how their placement and distribution support users and applications.
- *Expert Network*: information needed to accomplish a task. It includes how a particular task should be done as well as how some problems should be solved. Expert networks can be composed of manuals, specific procedures, and of course, experts.

Regarding to the hierarchical and informational networks, we should first analyse communicating flows inside the organisation. They can be:

- *Upward and downward*: these flows allow that communication through the hierarchical network occurs, and they are used as a command-reporting channel. That is, orders and reports between managers and subordinates are the main communicating flow. Downward vectors are associated to commands that flow from managers to their employees (hierarchical structure). The main objective of downward vectors is to ensure that every person in the organisation knows the organisation's goals and, in this way, uncertainty is reduced by increasing

credibility and reliability. Therefore, downward vectors will be used to communicate elicitation goals to organisation's personnel. Upward vectors are the channels where communication flows from employees to managers. The main goals of these vectors are commitment stimulation and identification. Then, upward vectors will be used to promote commitment when requirements are elicited and defined.

- Horizontal: these flows allow that communication among employees on the same hierarchical level occurs, and they are used for accomplishing routine teamwork. Horizontal vectors are established among peers or sectors. Vectors have the goal of improving organisational development by increasing internal cohesion, and facilitating management processes. In our proposal, horizontal vectors will be used to elicit non-conflicting requirements due participation of all stakeholders and easier circulation of messages.
- Transversal: these flows allow that communication among employees in different areas that are not connected through the hierarchical network occurs. Communicating between a project manager and its personnel (in different areas) is another frequent situation in which these flows appear.

In our empirical studies on several enterprises of different realms (Clinics, Newspapers, Hydroelectric enterprises) a sample of tasks that involves more than one area of each organisation was used to detect communicating flows or vectors, and to determine their predominant direction – upward/downward, horizontal, or transversal. After that, We define our strategy to conduct the requirements elicitation process [8,5].

As a first heuristic, we found that *when predominant vectors are horizontal and upward/downward, starting elicitation through organisation's areas is more convenient. On the other hand, when transversal vectors are significant, starting elicitation through organisation's processes is easier.*

Therefore, detecting vectors can help define how messages flow to perform a task and consequently who should be addressed. Analysts should know the vectors before starting the elicitation process. This knowledge influences the way the elicitation process is done. For example, enterprise behaviour is more committed or imposed depending on the predominance of horizontal or vertical vectors respectively. The latter situation, vertical vectors, would generate unreliable requirements.

On the other hand, if transversal vectors are predominant, procedures are more suitable of been produced by using different backgrounds and hence requirements quality is improved.

Figure 1 presents an example of the *information network* showing the information flows of the Maintenance Area of an organisation involved in controlling hydroelectric resources. Nodes in the network are not only people but also procedure manuals and machinery manuals provided by manufacturers. Modelling this network could produce an interesting approximation of where requirements of this area should be elicited.

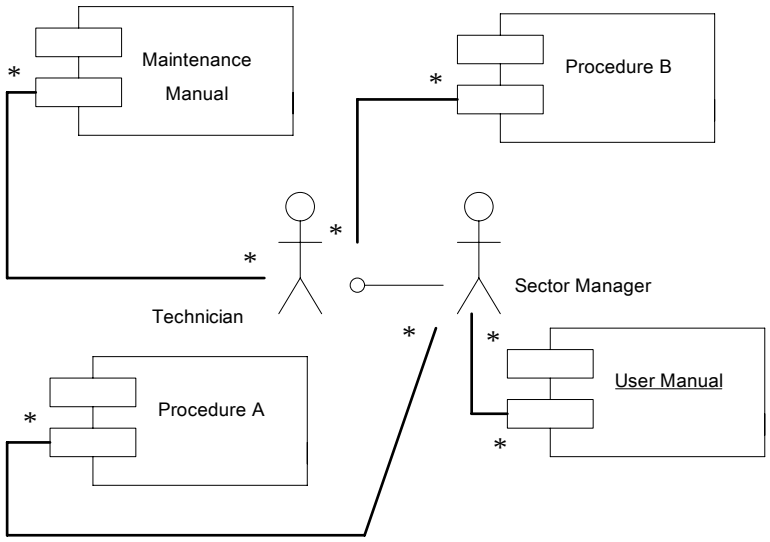

Figure 1: Information Network of the Maintenance Area

3 Improving Use Case Definition

There are several approaches to define and specify use cases [11,12,13]. In this section, we adopt the model defined by Cockburn [11] because it conceptually adds elicitation to use cases theory. This model defines the system and stakeholders at a generic level, meaning that the system under design is a mechanism to carry out a contract between various stakeholders. The use cases give the behavioural part of that contract. Every sentence in a use case might describe an interaction between two actors, or what the system must do internally to protect stakeholders' interests. To carry out its responsibility, the system formulates subgoals. It can carry out some subgoals internally. It needs the help of another, *supportin*g, actor to carry out others. This supporting actor may be a printing subsystem or it may be another organization, such as a partner company or government agency.

Additionally, *scope* is the word used by Cockburn for the extent of what we consider to be designed by us, as opposed to already existing or someone else's design job. *Functional scope* refers to the services your system offers. It will eventually be captured by the use cases. You are deciding the functional scope at the same time you are identifying the use cases. The two tasks are intertwined. *Design scope* is the set of systems, hardware and software, that you are charged with designing or discussing. Cockburn establishes the following design scope names:

- "Enterprise" scope discusses the behaviour of the entire organization or enterprise in delivering the goal of the primary actor. Business use cases are written at enterprise scope.
- "System" scope means just the piece of hardware/software you are charged with building. Outside the system are all the pieces of hardware, software and humanity that it is to interface with.

- "Subsystem" scope means you have opened up the main system and are about to talk about how a piece of it works.

The following subsections introduce our proposal for taking advantage of communication networks when producing use cases. Each following subsection details a particular phase of our approach.

3.1 Eliciting Use Cases

To elicit use cases, we firstly carry out a client-supplier analysis of actors. That is, each candidate actor of our Information Network plays the role of supplier of all other member of the network by informing all items supplied for each case. Then each candidate actor plays the role of client of all other member of the network by producing another list of received items.

Secondly, we chronologically order interchanges between suppliers and clients; and finally, we group all interchanges that belong to the same process. These steps allow us to systematically detect communication among stakeholders guided by the information network.

Then, by applying abstraction, every item supplied by a "supplier actor" is identified as a potential use case. Then, we analyze complexity determining whether further details are needed.

Let us introduce an example. Figure 3 shows the information network of the buying and selling processes of the Global Solutions Inc. organization, which trades mobile phones. Information is flowing, for example, from a supervisor to a seller or a sell agent by using the "Plan list", which is required by the selling process. Then, the seller produces a "SAV form" by filling information needed by an activator in order to initialize the mobile phone. Additionally, the seller produces a "Service form", which is received by an employee of the Administration Area. From this employee, the seller requests information needed to configure the mobile phone and he also uses the "Telephone user manual" as source of knowledge to accomplish the task.

Similarly, other flows of information are detected and they constitute the basement to build relationships in Figure 3. From this information network, and by performing a client-supplier analysis, we obtain information summarised in a tabular description. It is arranged according to the following concepts:

- *Process:* each process identifies a set of activities that an actor receives as client and a set of activities that this actor perform as a supplier.
- *Source*: represents the actor playing the role of supplier.
- *Target:* represents the actor playing the role of client.
- *Transaction:* it represents each activity performed by the source actor.
- *Type*: transaction type that can be internal – only involving actors internal to the system – or external – involving at least one external actor.
- *Execution order*: temporal order in which transactions must occur.

Here, it is important to note that these information must be only considered as a medium to organize elicited elements at a first stage. They will produce the first round of use cases, which will be refined later into more detailed information. Hence, we should not spend too much time and effort getting deeper details during this phase.

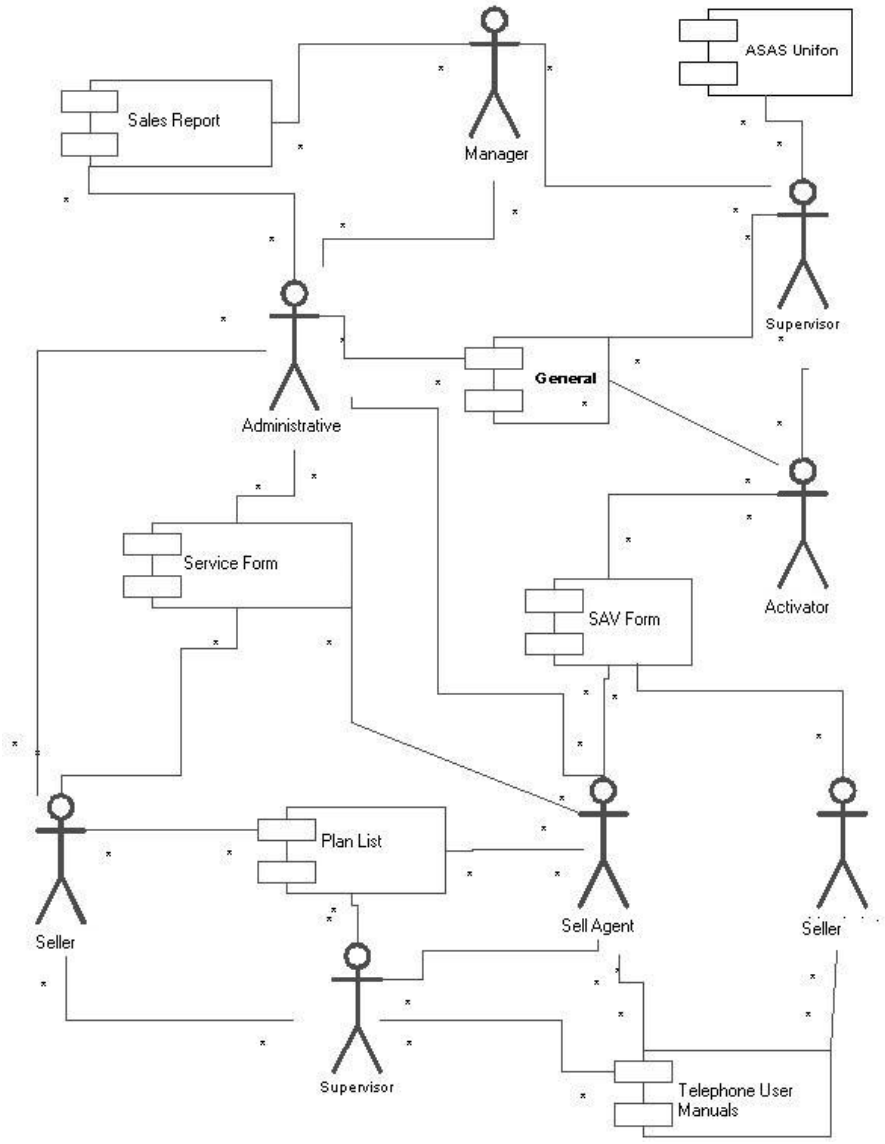

Figure 3: Information Network of the Global Solution Inc. organization

To further detail use cases, we start specifying contextual use cases [11] by selecting external transactions that represent interactions between the system and its context. To do so, we suggest the following heuristics.

- External actors can only receive or send information to the system. Then, transactions produced by an external actor should be considered as *information flows* supplied to the system by this external actor.
- Internal actors can request the execution of functions. Then, transactions produced by an internal actor should be considered as *functionality* to be described by the use case.

3.2 Building Use Cases

To build use cases, we fill the Cockburn templates by iteratively refining descriptions according to

the template's requirements. However, information to build use cases are now derived from the information network summarised in the client-supplier table. This information has been reviewed by members of the expert network, who might contribute with further details – such as extend relations that could complement the cases.

For example, Figure 4 shows a process where the actor "Administrative" initiates the use case "Perform Payment". Resulting products are received by the external actor "BsAs Collect", which initiates the use case "A.Op.Correct" if there are non-approved operations, and additionally producing a list of them. The actor "Administrative" initiates the use case "Conciliate" that delivers a list of deposits to the actor "BsAs Collect".

The contextual use case "Perform Payment" represents how to process information of payments. First of all, an Administrative sends information of sells to his headquarters. This information is analysed to detect missing data, which are requested by BsAs. Collect along with a deposit form. There are also possible extensions, such as reimbursements or detection of incorrect data, which are specified through additional use cases. Finally, every use case description details the requirement's source in order to improve trace ability.

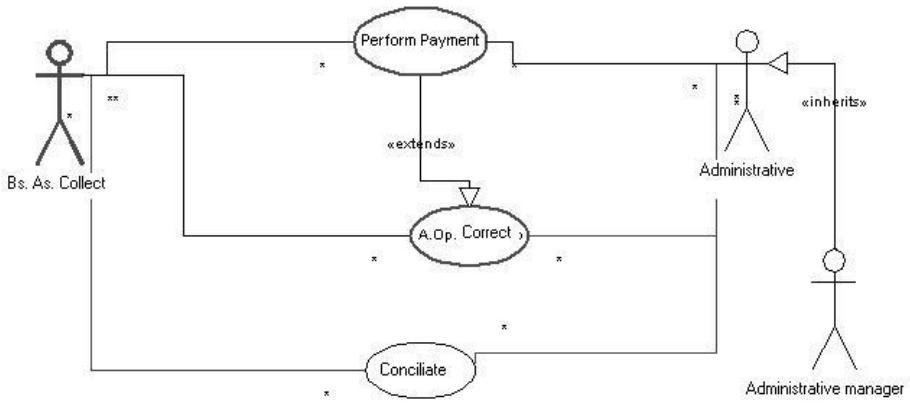

Figure 4: External and internal actors of a payment process

Table 1 partially shows client-supplier information used to build the "Perform Payment" use case.

Execution Order	Type	Source	Destination	Transaction
1	EXT	Administrative Manager	Bs.As. Collect	Operations analysis
2	EXT	Bs.As. Collect	Administrative	Claim
3	EXT	Bs.As. Collect	Administrative	Observations
4	EXT	Administrative	Bs.As. Collect	Correct Operations
5	EXT	Administra-	Bs.As. Collect	Conciliation

		tive		

Table 1: Client-supplier table for the Perform Payment process

3.3 Additional Remarks

Our strategy has been successfully applied to elicit software requirements of a Telecommunication Consulting Company – Global Solutions Inc. – which is a partner of Telefónica de Argentina Co. This organization is characterized by its informal structure, where domain knowledge mainly relies on people. Using communication networks helped us to find appropriated stakeholders, particularly those whose skills and background better contributed to describe the organization's areas. Besides, eliciting through communication networks allowed us to verify requirements' correctness through validation of external and internal users.

Using the client-supplier table allowed us to clearly understand existing processes and their transactions. By combining communication networks and the table as supporting tools, we got enough information to produce use cases reducing ambiguity. However, note that our proposal does not reduce natural language ambiguities, which might hinder the process whether description is not complemented with sequence and activity diagrams.

4 Conclusions

Gathering the right requirements is still an ongoing challenge for requirement engineering. Our proposal aims at reducing time and effort when acquiring domain knowledge, and helps address information sources of true requirements. Using the client-supplier table introduces an approach to understand and improve organizational processes. This fact indirectly contributes to improve use case definition, as we have illustrated. However, our proposal needs further validation to quantify time reduction and understanding. On this line, we are defining some metrics on understand ability of use cases, which will give a more precise indicator to evaluate advantages.

Acknowledgments

This work is partially supported by the UNComa project 04/E059, and developed as part of a Master Thesis supervised by Alejandro Oliveros. Also, many thanks to Alejandra Cechich for her comments on the draft of this paper.

References

[1] Christel M., and Kang K., 1992. Issues in Requirements Elicitation. *Technical Report, CMU/SEI 92-TR-12.*

[2] Al-Rawas A., and Easterbrooks S., 1996. Communication Problems in Requirements Engineering. A Field Study. In *Proceedings of the First Westminster Conference on Professional Awareness in Software Engineering,* London.

[3] Robinson W., and Pawlowski S., 1999. Managing Requirements Inconsistency with Development Goal Monitors. In *IEEE Transactions on Software Engineering, Vol. 25, No. 6, 816-835.*

IEEE Press.

[4] Ciamberlani L. And Steimberg L., 1999. *Comunicación para la Transparencia*, Granica Ed.

[5] Luzuriaga J., Martínez R., Cechich A., Managing Enterprise Communication Networks to Improve the Requirements Elicitation Process", ICEIS 2002, In *Proceedings of the 4th International Conference on Enterprise Information Systems*, Ciudad Real, Spain, 3-6 April 2002, ISBN 972-98050-6-7 (770-775).

[6] Luzuriaga J., Cechich A., and Martínez R., Classifying Requirements Perception to Guide Exploratory Processes in Requirements Elicitation, *V Workshop Iberoamericano de Ingeniería de Requisitos y Ambientes de Software*, IDEAS 2002, La Habana, Cuba, 23-26 April 2002.

[7] Harrington H., 1993. *Mejoramiento de los Procesos de la Empresa*, Mc Graw-Hill.

[8] Luzuriaga J., and Martínez R., 2001. Recopilación de Información en un Proceso de Auditoria de Normas de Calidad, *Reporte Técnico 010, Universidad Nacional del Comahue,* Argentina.

[9] Beyer and Holtzblatt, 1998. *Contextual Design: Defining Customer-Centred Systems*, Morgan Kaufmann.

[10] Rumbaugh J., Jacobson I., and Booch G., 1999. *The Unified Modeling Language Reference Manual*, Addison-Wesley.

[11] Cockbur, Alistair "Writing Effective Use Cases " , ISBN 0201702258, 2000

[12] Davis Alan, Hickey Ann, Elicitation Technique: How Experts Do It ?, *In Proceedings of the 11th IEEE International Requirements Engineering Conference*, 2003

[13] Nuseibeh, B., Easterbrook, S. Requirements engineering: a road map, In *Proceedings of the International Conference on Software Engineering*, ICSE 2000, Future of SE Track 2000 (2000) 35-46