A Requirements Engineering Process extended to Context Information Management

Viviana Castelli
School of Computer Science, National University of La Plata
La Plata, Argentina
castelli_viviana@hotmail.com

Rodolfo Bertone
Institute of Research in Computer Science III-LIDI
School of Computer Science. National University of La Plata
La Plata, Argentina
pbertone@lidi.info.unlp.edu.ar

Pablo Thomas
Institute of Research in Computer Science III-LIDI
School of Computer Science. National University of La Plata
La Plata, Argentina
pthomas@lidi.info.unlp.edu.ar

Alejandro Oliveros
School of Computer Science, National University of La Plata
La Plata, Argentina
aoliveros@gmail.com

Abstract—**Context-Aware Software Systems adapt their response to the execution context. Therefore, context information processing is fundamental, since it determines the behavior of the application. Thus, a mechanism for the collection, representation and validation of context information should be provided. In this paper, an extension of Loucopoulos' requirements engineering process incorporating context information processing is presented. Two real-world study cases from the home and building automation fields of application are presented.**

Key words: Context-Aware Systems, Requirements Engineering, Context, Context Elements, Home Automation, Building Automation

I. INTRODUCTION

Context-aware systems are an emerging arm of software systems that, although young, is widely accepted by several fields of application, such as home and building automation and e-commerce, among others.

The main feature of context-aware systems is that they can constantly adapt to what happens in the application environment; that is, the response of the system changes according to the changes that occur in its context.

The need to apply a Requirements Engineering process is analyzed and studied by the Software Engineering community for this type of systems [1].

Various Requirements Engineering approaches for context-aware systems have been analyzed ([2], [3], [4], [5], [6] and [7]), and it has been noticed that this field of study is not fully mature yet.

In order to obtain a Requirements Engineering process for context-aware systems, [8] and [9] propose the definition and use of a set of concepts that allow establishing a uniform and concise representation of contextual information. The "context" of an application is formed by the application environment entities and the information that characterizes them, entities being objects, users, time, or services, among others. Additionally, a context taxonomy has been defined that allows classifying context entities into different types and which supports context information elicitation.

Later, [10] and [11] created new concepts that allow establishing the component parts of an application context information. These parts are: context element, context attribute, and context attribute value.

A context element represents an entity that is within the system's physical environment and collaborates or interacts with it. A context attribute is a measurable feature of the context element. Lastly, the context attribute value is the result of measuring the corresponding attribute.

Articles [10] and [11] also propose a representation that allows specifying all the information of a context element in an only repository.

In this sense, in order to capture the essence of the system, the differences between user requirements and system requirements, as proposed in [12], should be considered.

According to [12], user requirements express properties of the domain or business process that the user wishes to have in the new system. On the other hand, system requirements express system properties or abilities that will allow complying with at least one user requirement. Therefore, each user requirement can be related to one or more system requirements and vice-versa, this difference being essential.

Below, in Section 2, a Requirements Engineering process extended to context information treatment is proposed. Then, in Section 3, two study cases where the process was applied are described. In Section 4, the conclusions are presented, and Section 5 includes future works.

II. Requirement Engineering Process extended to Context Information Treatment

In order to build the process for the contextual information management, it was decided to amplify an existing process requirement model, which is the model presented by Loucopoulos and Karakostas [13], due to the fact that it is a stabilized model, widely acknowledged by the Requirement Engineering community, still used, for instance, to classify the advances in the Requirement Engineering field [17]. Nevertheless, the main reason for this choice is that it is a model with a high degree of abstraction and generality very suitable to approach a new field as the requirement field is regarding contextual information.

Context-aware systems have user needs that are directly related to the occurrence of specific events in the context of the application. That is, upon a change in the context, the system must comply with some functional requirement and some non-functional requirement. For instance, a user needs his cell phone to send a text message to inform him when he is near a given restaurant. For the system to be able to meet this user requirement, it needs to know its current location. Consequently, context information identifying the location of the user is a precondition to send the text message. Additionally, the system should inform as soon as possible based on user location, otherwise the message would be outdated. There is therefore a limitation in relation to the time the system takes to communicate with the user. In this case, this is a non-functional requirement.

For this reason, we propose extending the process described in [13] in order to consider context influence in requirement elicitation, taking new emerging elements into account. It is essential that context information that is relevant for the application be obtained, documented and validated, since any incorrect information will render the system unsuitable for the real needs of the user.

As a consequence, the elicitation, specification and validation of context elements that are inherent to a context-aware system are incorporated to the Requirement Engineering process as well as the interactions with the activities of the original process model.

Figure 1 presents a graphic representation of the process described in [13] with the extensions proposed in this paper.

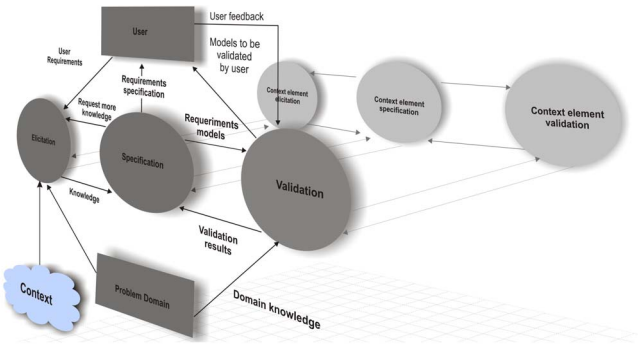

Figure 1. Requirements Engineering process including the elicitation, specification and validation of context elements.

A. *Context Elements Elicitation*

According to [13], requirement elicitation consists in obtaining the relevant knowledge needed to produce a model of a problem's requirements.

Consequently, the first stage of the extended process consists in identifying all context information that is relevant for the system to be built, particularly those context elements that can affect the behavior of the system.

In a context-aware system, and due to the close relationship between user requirements and context information, context elements, as well as its attributes and values, appear during the requirement elicitation stage. To obtain context elements, the application context information required to fulfill each user requirement has to be determined.

Therefore, context elements elicitation is an activity that should be carried out during the elicitation sub-process defined in [13].

The analyst should determine the context elements and their corresponding attributes and values, decide if the information obtained is relevant for the operation of the system, and understand the meaning of each context element and its relation with one or more requirements.

Traditionally, the output of the elicitation process described in [13] is knowing the domain of the problem. Part of this knowledge comes from user needs and objectives in relation to the system [1]. The objective of this paper is specifically identifying user requirements or needs, as well as producing context elements that include a reference to the user requirement generating them. Figure 2 shows a diagram of the sub-process proposed. In the first place, it is proposed to make non-structured interviews to experts to obtain knowledge domain and then from this knowledge, create a questionnaire. Then, Figure 3 represents the list of context elements and their corresponding attributes and values.

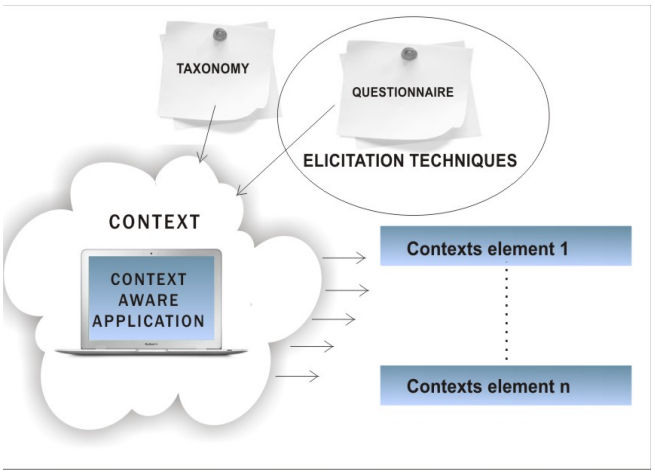

Figure 2. Context elements elicitation sub-process.

User Requirement 1:			
Context element	**Context element**	**Context attribute**	**Value**

.....

User Requirement n:			
Context element	**Context element**	**Context attribute**	**Value**

Figure. 3. List of context elements.

B. Context Elements Specification

According to [13], a specification can be considered as a contract between users and developers that defines the desired functionality, but does not indicate how this functionality is achieved.

Once the context elements elicitation stage is completed, the knowledge obtained has to be documented. The specification stage consists in organizing, synthesizing and "translating" the information obtained during the elicitation process.

This is the central stage of the Requirements Engineering process. If during the creation of the specification there is a need for deeper knowledge about the application context, the context elements elicitation process is triggered again, generating an iteration of the process that allows improving the obtained knowledge.

In [13] there is no distinction between user requirements and system requirements. However, the authors in [12] emphasize the difference between these two types of requirements and the importance of documenting them. In this paper, we incorporate the concepts explained in [12] to the process described in [13] and apply the end result to context-aware systems.

Each user requirement is mapped to one or more system requirements. Therefore, each system requirement is based on the originating user requirement.

To specify each context element, a diagram as the one shown in Figure 4 is used, where each context element is linked to the originating user requirement and the system requirement that is generated as a response.

Context element #:	
User requirement	
System requirement	
Context type	
Context element	
Attribute or feature	
Value	

Figure. 4. Representation of the diagram used to specify context elements.

Figure 5 summarizes the context information specification sub-process.

C. Context Elements Validation

In [13], requirements validation is defined as the process that certifies that the requirements modeled are consistent with the intentions of customers and users.

Therefore, the final stage of the process allows ensuring that the knowledge acquired about the application context information is suitable to fulfill system requirements.

During this stage, each context element is validated based on the associated system requirement. To do this, application requirements have to be previously validated. If any requirement is removed, it is also removed from the associated context elements. Additionally, the inclusion of new system requirements as a consequence of the validation process may generate the consideration of new context elements.

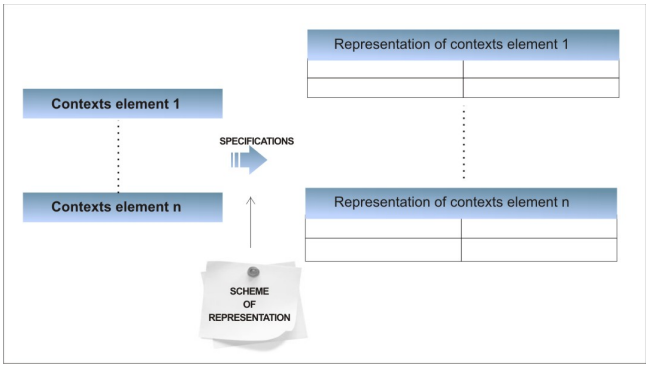

Figure. 5. Context elements specification sub-process.

This validation phase is non-structured, and allows identifying and correcting errors in the context information obtained before actually developing the software, thus helping avoid the higher costs of correcting them at a later stage.

III. STUDY CASES

Home and building automation are two context-aware systems areas that have grown significantly in recent years. Their assessment is of interest due to the amount of context information that can be managed [9].

According to [14], building automation is the incorporation of automated technical management systems to equipment used in industrial or commercial buildings (offices, corporative buildings, hotels, companies, and the like), to reduce energy consumption and increase comfort and security.

According to [15], home automation, also referred to as "Domotics," includes the set of solutions that, by means of available techniques and technologies (electricity, electronics, computers, robotics, telecommunications), help improve the use, management, and control of aspects related to the home (comfort, security, reduced energy consumption, communications, computers, television, among others).

The company Digital Home [16], which implements home and building automation systems in the city of La Plata, Argentina, incorporated in 2009 the use of the extended process suggested in this paper as part of a plan to improve its Requirement Engineering tasks. In the following section, two study cases that were carried out in collaboration with the company are presented. In both cases,

a specific elicitation questionnaire based on the problem to solve was used.

A. First study case: Automation solution for a hotel

Upon receiving a request for a building automation solution, the company contacted the interested customer to put the process proposed into practice. The customer requested the incorporation of a software system in the hotel to automate certain activities. Among other features, the hotel needed to minimize its energy consumption in common areas with enough natural lighting and use alarms that allowed detecting intrusions and thus improve hotel security.

Context elements elicitation: The customer was interviewed to apply a questionnaire designed for building automation systems. Then, based on the questions and their corresponding answers, context elements were determined. Each question/answer was analyzed to determine context elements with their corresponding context attribute and value linked to the user requirement emerging from that question. That is, the context elements needed to fulfill the user requirement were determined using the context taxonomy proposed in [8] and [9] to guide the elicitation process. Figure 6 below presents a small part of the questionnaire used.

1- EFFICIENT LIGHTING MANAGEMENT

a. Do you want to automatically turn the lights on when there is insufficient light if human presence is detected in common areas? Yes/No.

Multiple use room, swimming pool, lobby, restaurant, etc. Indicate those that apply:

Figure. 6. Part of the questionnaire applied to building automation.

Figure 7 presents, as the result of the elicitation process, one of the user requirements together with the derived context elements, where the values of each context attribute are subject to user response.

User Requirement 1: Optimize energy use at the hotel			
Context	**Context element**	**Context**	**Value**
1	Presence sensor	Operation	Works/Does not work
2	Environment	Lighting	Sufficient/Insufficient
3	Light sensor	Operation	Works/Does not work
4	Person	Presence	Yes/No

Figure. 7. Possible context elements.

Context elements specification: considering the limited example presented in Figures 6 and 7, the user requirement generated at least two system requirements. Figures 8 and 9 show two context elements specification examples.

Context elements validation: the system requirement associated to each context element obtained is analyzed, and the relationship defined with the context element is checked for correctness. To do this, the attribute value represented by the context element in validation is checked to determine if it is a pre-condition for the fulfillment of the system requirement and, therefore, for the user requirement originating it.

In the limited example presented, the operation of the presence sensor represents a pre-condition to be able to determine the presence of people in common areas. Similarly, the operation of the light sensor is necessary to determine the level of light in the environment.

Context element #: 1	
User requirement	Optimize energy use at the hotel
System requirement	Determine the presence of people in common areas.
Context type	Object context
Context element	Presence sensor
Attribute or feature	Operation
Value	Works/Does not work

Figure. 8. Context element specification example.

Context element #: 2	
User requirement	Optimize energy use at the hotel
System requirement	Determine the level of natural light in common areas
Context type	Object context
Context element	Light sensor
Attribute or feature	Operation
Value	Works/Does not work

Figure. 9. Context element specification example.

B. Second study case: Home automation solution

In this case, the customer is a citizen of the Autonomous City of Buenos Aires, Argentina, who owns the property to automate. Among other features, the customer requested that the system can simulate human presence, the use of alarms to detect fires, and the automation of some tasks that allow increased home security.

Context elements elicitation: First, a questionnaire for home automation systems was designed, and, as in the previous study case, it was filled during a meeting with the customer. Figure 10 below shows part of the questionnaire used.

Figure 11 shows some of the context elements obtained from the questionnaire, with the values of each context attribute being determined by the answer of the user.

1- PRESENCE SIMULATION

a. Do you want to automate certain events through the software system (turning lights on/off, opening/closing blinds, turning on the sound system or a TV) by day and time, in order to create the appearance of human presence? Yes/No.

What do you want to automate? Lights, blinds, sound system, TV. Indicate those that apply.

Simulation time:

Figure. 10. Part of the questionnaire applied to home automation.

Context elements specification: considering the limited example presented, the user requirement of *simulating presence at home* was translated into the system requirements necessary to fulfill it, and their subsequent representation in the previously presented diagram. Figure 12 shows the specification of one of the associated context elements.

User Requirement 1: Simulate presence at home			
Contex	**Context element**	**Context attribute**	**Value**
1	Electric circuit	Status	Off/On
2	Blinds	Status	Open/Closed
3	Blind motor	Operation	Works/Does not
4	TV	Operation	Works/Does not
5	TV	Status	Off/On

Figure. 11. Possible context elements.

Context element #: 1	
User requirement	Simulate presence at home
System requirement	Opening and closing blinds
Context type	Object context
Context element	Blinds
Attribute or feature	Status
Value	Open/Closed

Figure. 12. Context element specification example.

Context elements validation: the system requirement associated to each context element obtained was analyzed, and the relationship defined with the context element is checked for correctness.

In this limited example, the status of the blinds (open or closed) has to be determined to apply the corresponding necessary action (closing or opening them). It makes no sense trying to open a blind whose current status is "open."

IV. Conclusions

This paper is the continuation of [8], [9], [10] and [11], where, after analyzing the state of Requirement Engineering for context-aware systems, the concepts needed to improve the understanding and specification of application contexts were introduced.

This proposal extends the Requirement Engineering process presented in [13] by incorporating the elicitation, specification and validation of context elements, improving the first stage in the development of context-aware software systems.

The extension proposed is based on the link between the requirements of the users of an application and the context information at each stage of the requirements process. Also, the relation and differences between user requirements and system requirements is analyzed, and how the latter are needed to be able to fulfill each of the former.

Context information is represented by context elements (with their corresponding context attributes and values). Thus, from the extended process, there is a set of tasks to capture, specify and validate context elements, user requirements and system requirements; this information is essential to manage the development of a context-aware application.

For the specification stage, an extension of the context elements representation scheme proposed in [10] was used. This scheme includes a reference to the user requirement that originates each context element, and the generated system requirement was added.

The validation of the context elements obtained is done by validating application requirements.

As proof of concept, two real-world study cases from the home automation and building automation areas that applied the tasks proposed for the process were analyzed. To elicit context elements, two ad-hoc questionnaires oriented to home automation and building automation, respectively, were used, together with the context taxonomy defined in [8] and [9].

The context elements obtained allowed clearly establishing under which environment conditions the systems should modify their response. Each context element uses its attribute value to identify a possible change in system response. Also, the actions that the application would perform as a consequence of the change in context defined by each specific element were determined.

The use of the extended process allowed obtaining, representing and debugging context information based on user requirements.

Finally, it should be mentioned that, by testing the process in real-life situations, the usefulness of the proposal has been verified.

V. Future Work

To continue with this research line, heuristics will be defined to obtain context elements from user requirements. These heuristics will be a set of rules to be used during the context elements elicitation activity.

Finally, a software requirements specification standard will be broadened to include context elements.

References

[1] Thomas Pablo, Oliveros Alejandro, Elicitación de Objetivos a partir de Escenarios, VIII Workshop on Requirements Engineering WER'05, Porto, Portugal, June, 2005.

[2] Finkelstein, Savigni, A Framework for Requirements Engineering for Context-Aware Services, First International Workshop from Software Requirements to Architecture-23d International Conference on Software Engineering, 2001.

[3] Godbole, Smari, Human perspective based context acquisition, learning and awareness in the design of context aware systems, Department of Electrical and Computer Engineering, University of Dayton, Military Communications Conference, 2006. IEEE.

[4] Hong, Chiu, Shen, Requirements elicitation for the design of context-aware applications in a ubiquitous environment, Proceedings of the 7th international conference on Electronic commerce, 2005.

[5] Kammanahalli, Gopalan, Sridhar, Ramamritham ,Context Aware Retrieval in Web-Based Collaborations, Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops (PERCOMW'04).

[6] Kolos, Mazuryk, Poulisse, van Eck, Requirements Engineering for Pervasive Services, October 16, 2005. 2nd Workshop on Building software for pervasive computing.

[7] Seyff, Graf, User-Driven Requirements Engineering for Mobile Social Software- Software Engineering 2010 –Workshopband (inkl. Doktorandensymposium) 22.–26. February 2010. Paderborn, Germany.

[8] Castelli, Thomas, Bertone. "Ingeniería de Requerimientos para Sistemas Sensibles al Contexto, un estudio comparativo". XIV Congreso Argentino de Ciencias de la Computación (CACIC 2008)

[9] Castelli, Thomas, Bertone. Ingeniería de Requerimientos para Aplicaciones Sensibles al Contexto: "Un Procedimiento de Identificación y Representación de Elementos de Contexto". XV Congreso Argentino de Ciencias de la Computación (CACIC 2009).

[10] Castelli, Tesina de Grado para obtener la Licenciatura en Sistemas: "Definición de un Esquema de Representación de Elementos de Contexto. Su aplicación en Domótica". Facultad de Informática. UNLP. June 2010.

[11] Castelli, Thomas, Bertone, Captura, representación y validación de información contextual para Sistemas Sensibles al Contexto-XVI Congreso Argentino de Ciencias de la Computación (CACIC 2010), Morón, Argentina, 2010.

[12] Maiden, User Requirements and System Requirements- IEEE-March/ April 2008 ISSN: 0740-7459.

[13] Loucopoulos P., Karakostas V.; "System Requirements Engineering"; McGraw-Hill International series in Software Engineering, 1995.

[14] http://es.wikipedia.org/wiki/ October 2010.

[15] www.cedom.es, Asociación Española de Domótica (Spanish Home Automation Association).

[16] http://digitalhomeweb.com.ar/ October 2010.

[17] Hansen, S., Berente N. and Lyytinen, K., (2009) "Requirements in the 21st Century: Current Practice & Emerging Trends," in Lyytinen, Loucopoulos, Mylopoulos, Robinson eds, *Design Requirements*

Engineering: A Ten-Year Perspective, Springer-Verlag (Lecture Notes in Business Information Processing Series Vol.14), 2009.