Modelling CRM Implementation Services with SysML

Luis H. Bibiano, J.A. Pastor, Enric Mayol

Universitat Politecnica de Catalunya
{lbibiano ,pastor, mayol}@lsi.upc.edu

Universitat Oberta de Catalunya
jpastorc@uoc.edu

Abstract.

CRM information systems are valuable tools for enterprises. But CRM implementation projects are risky and present a high failure rate. In this paper we regard CRM implementation projects as services that could be greatly improved by addressing them in a methodological way that can be designed with the help of tools such as SysML. Here we introduce and comment on our first experience on the use of SysML language, not very well known, for modelling the elements involved in the CRM implementation processes included in our method.

Introduction

Customer Relationships Management (CRM) systems are nowadays a complementary and valuable tool for enterprises that allow them to manage and identify valuable customer information for their market-oriented business processes (Bibiano and Pastor, 2006). Implementation services for these systems have been an interesting topic of discussion over the last years, due to rather negative results related to real CRM implementation processes as well as a lack or public CRM implementation methods.

Implementation projects for enterprise information systems such as ERP, CRM or SCM systems are quite different and possess specific characteristics. These affect the expected outcomes depending on the type of system, so the implementation approach and processes used should be adapted for each situation (Bibiano et al., 2007).

This justifies research on appropriate methods for addressing implementation services related to theses systems. Along this line, the authors have been working on the design of an adaptive method for the implementation of CRM system. This will hopefully help in the future reduction of the many risks and failures associated with the implementation services of these types of enterprise information systems.

In our method analysis and design effort, we have gathered some conceptual tools related to project management best practises as well as software engineering, with the intention of designing a sound method for the implementation services of CRM. For this purpose, in this paper we introduce our early and tentative experience on the use

of a recent modelling language called SysML, which we believe to be not very well known, and that is helping us in the design of graphical representations of the CRM implementation processes included in our method.

Service engineering is an emerging topic within the recent so-called Service Science, Management and Engineering wave, where there is not yet a formal and comprehensive approach to the design and engineering of service systems in their full complexity. After our first experiences, we believe that the SysML modelling language, that extends and adapts UML to more general systems modelling, appears as an interesting alternative to engineering in an integrated manner some of those complex service systems.

From our understanding of, and experience with, implementation projects related to enterprise information systems we regard CRM implementation as services that could be greatly improved by addressing them in a methodological way that can be designed with the help of tools such as SysML. Along this line, the goal of this paper is to show the possibility of the application of such a comprehensive modelling language in the conceptual and graphical description of CRM implementation elements. Beyond this viability exercise, here we do not pretent to compare SysML with other more specialized modelling languages, such as BPMN, IDEF-X or the like. Neither we pretend to compare SysML with its parent UML in representation power regarding modelling implementation methods. However, we do pretent to benefit from SysML in its attributed advancements in comparison with these other notations with regard to general systems modelling. As well as to apply it to modelling CRM implementation services which can be naturally regarded as complex working systems.

In the rest of the paper we first present the concepts around CRM systems and their implementation. Then we introduce SysML and we discuss its application to modelling CRM implementation services. We then show examples of artefacts drawn from our use of SysML for modelling our CRM implementation method. After some comments, we conclude with some reflections of future uses of SysML in our immediate context.

1. – CRM Systems and their Implementation

A Customer Relationship Management (CRM) system includes tools, technologies and procedures that allow managing, improving and facilitating sales, support and related interactions with customers, prospects, and business partners throughout the enterprise (Davenport et al., 2001). The main purpose of a CRM system is to help sales and marketing people to analyze of customer behavior and its value for the organization by using technology and human resources.

Research already made in this area states that, by using CRM systems, organizations may obtain loyal and more satisfied customers by improving their business, reducing

acquisition costs and gaining acknowledgment of their brand, which it is translated to a better financial performance (Gefen & Ridings, 2002). Some organizations only implement some functionalities of a CRM system and, although this action may limit the whole system functionality, it is a great step towards integrating the missing modules in the future to achieve a complete implementation.

CRM implementation projects are classified as highly risky projects. We have analyzed the problems arising in CRM implementations, and this has led us to consider the possible causes and to analyze the current use of implementation methods to deploy CRM and other types of enterprise information systems (EIS) in organizations (Bibiano et al, 2007). We believe that a main cause for the high failure rate and high risk in EIS implementations, and even more for the CRM case, can be the inappropriate use of implementation approaches and methods, inherited and only slightly adapted from ERP implementation methods.

Given the many differences between CRM and prior ERP systems, it seems to us that this may be a basic source of problems. Thus, we think that each one of the EIS types should be addressed in a different way, according to the particular nature of the system type and project.

Moreover, it seems that EIS implementations should include certain features not considered until now in ERP projects, that could aid and support the overall process in order to obtain a better control of their development. For this purpose, we think that such features ought to be gathered from the analysis of other related areas. Looking into the software engineering area, we have found agile methods to be more apt for the case of implementing CRM systems (Bibiano et al. 2007).

Agile methods are a recent family of software development processes that focus on customer satisfaction, low level failure rates, fast delivery times and answers to rapidly changing environments. They point to minimize possible risks by using defined period of times called iterations, which can be viewed as “mini-projects” that include the required tasks to create parts of the whole project (Boehm, 2003). CRM systems are always projects whose application needs to be prepared to quickly respond to changes in requirements, maybe one of the reasons being the dynamic and sometimes unstable environment of most commercial processes in organizations, specially those of a more analytical or decisional nature. For this reason, in CRM projects requirements are changing and unpredictable at initial stages and they are discovered and stabilized along the whole implementation process. These are some reasons why agile approaches appear to be more suitable for the implementation of CRM systems, in contrast with the implementation of other types of enterprise information systems, such as SCM or ERP systems (Bibiano et al., 2007).

2. - SysML modeling language

The OMG systems Modelling Language (OMG SysML) is a general-purpose graphical modelling language for specifying, analyzing, designing, and verifying complex systems that may include hardware, software, information, personnel, procedures, and facilities. In particular, the language provides graphical representations with a semantic foundation for modelling system requirements, behaviour, structure, and parametric, which is used to integrate with other engineering analysis models (House, 2006). SysML represents a subset of UML 2 with extensions needed to satisfy the requirements of the UML™ for Systems Engineering RFP.

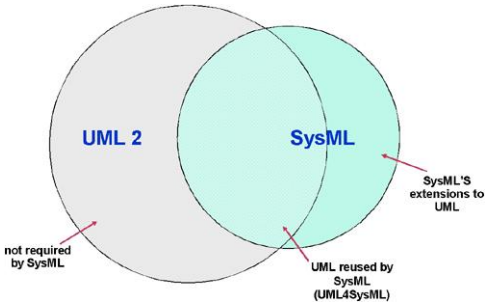

Fig. 1. SysML and UML

The block is the basic unit of structure in SysML and can be used to represent hardware, software, facilities, personnel, or any other system element. The system structure is represented by block definition diagrams and internal block diagrams. A block definition diagram describes the system hierarchy and system/component classifications. The internal block diagram describes the internal structure of a system in terms of its parts, ports, and connectors. The package diagram is used to organize the model.

Behaviour diagrams include the use case diagram, activity diagram, sequence diagram, and state machine diagram. A use-case diagram provides a high-level description of functionality that is achieved through interaction among systems or system parts. The activity diagram represents the flow of data and control between activities. A sequence diagram represents the interaction between collaborating parts of a system. The state machine diagram describes the state transitions and actions that a system or its parts perform in response to events.

SysML includes a graphical construct to represent text based requirements and relate them to other model elements. The requirements diagram captures requirements hierarchies and requirements derivation, satisfy and verify relationships that allow a modeller to relate a requirement to a model element that satisfies or verifies the requirements. The requirement diagram provides a bridge between the typical requirements management tools and the system models.

The parametric diagram represents constraints on system property values such as performance, reliability, and mass properties, and serves as a means to integrate the specification and design models with engineering analysis models. SysML also includes an allocation relationship to represent various types of allocation, including allocation of functions to components, logical to physical components, and software to hardware.

3. - SysML for modelling CRM implementation services

3.1 Issues about using SysML in the design of implementation methods

In section 2, we mentioned that the SysML diagrams can be used for the overall definition of the process of a project, since the language allows the inclusion of some items that are present in this type of work. SysML has been used as a tool in projects apart from the software area, such as aeronautics, electronics or telecommunication systems, but so far it has not been used in engineering services such as implementation projects. Thus, the application of SysML in the definition of an implementation method is an opportunity to may extend its range of use and to show a modelling alternative that groups characteristics from other modelling languages.

The implementation of CRM systems requires a set of processes that are defined according to the project's needs, including all its items (roles, activities, tasks, builds, requirements) that work together towards the design of the required method. Using the diagrams provided by SysML we have defined graphically some processes that are included in our tentative CRM implementation method; with the purpose of clarify the course of action to follow during the phases and builds of the project.

3.2 Modelling a CRM implementation method with SysML: Some examples

We have been working on the design of an implementation method suited for CRM systems by using some tools related to software development, one of it being the Open Unified Process (OpenUP). The method structure shares some elements of OpenUP such as the four great phases they propose for the creation of software projects (Inception, Elaboration, Construction, and Transition, already appearing in RUP) and some of their stages and tasks. In order to show the application of SysML in this method design, we present the modelling of important elements within our Elaboration phase.

The Elaboration phase is, chronologically, the second grand phase in the whole method. In this phase, the course of actions to take for most processes of the project are defined based upon the results of the analysis and the Project plan detailed in the Inception phase. The Elaboration phase defines the required build and tasks related to iteration planning, iteration control, risk management and architecture definition. In one phrase, Elaboration establishes the basis for the application and management of

the iterations that will be used in further activities of the method. Using the SysML language, the modelling of such elements is presented in the next paragraphs.

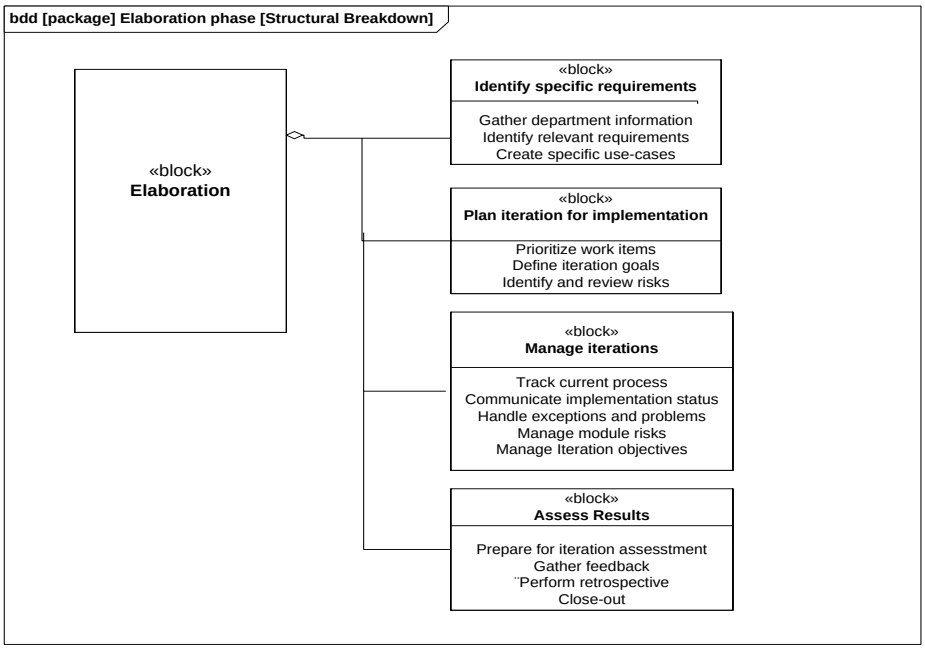

Fig. 2 CRM implementation Elaboration (block definition diagram)

The Elaboration phase's structure is shown in Figure 2, using a *Block definition diagram* with the four main activities of this phase, containing their main tasks. The *Internal Block Diagram*, shown in Figure 3, presents the flow of the inputs and outputs of the activities, as well as the outputs of the Elaboration phase.

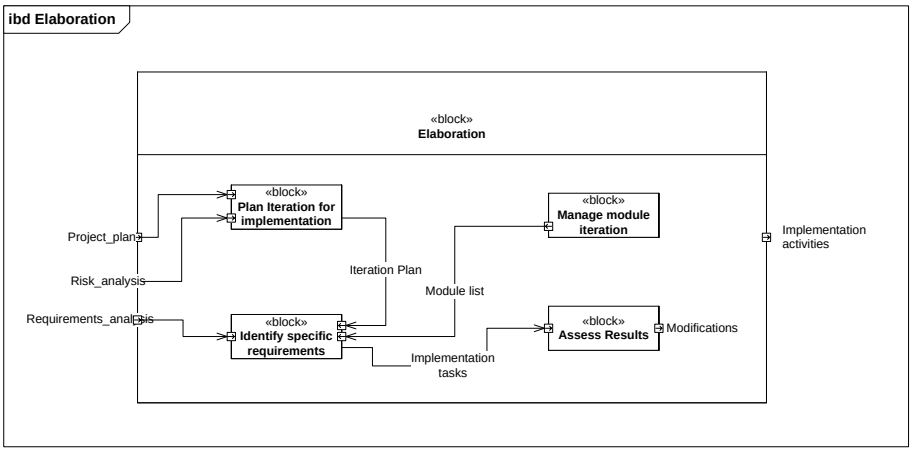

Fig. 3 CRM implementation Elaboration (internal block diagram)

Requirements diagram possesses a similar structure to the block diagram, showing the requirements for the Elaboration phase and its relationships. In Fig. 4, we have only shown a part of the whole requirements diagram, because of space matters.

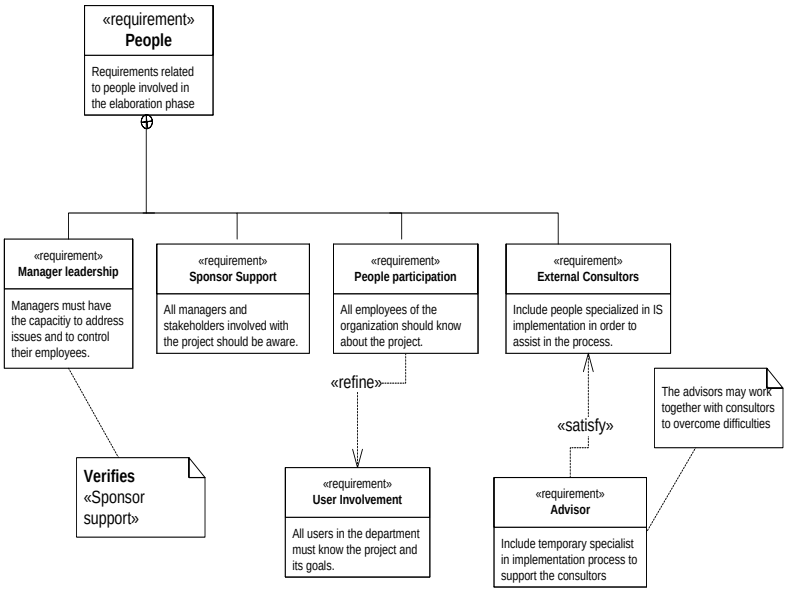

Fig. 4. CRM implementation Elaboration (requirements diagram)

The SysML language also includes a *Use Case diagram* that depicts the roles involved in the processes and the activities in which they are committed, as shown in Figure 5.

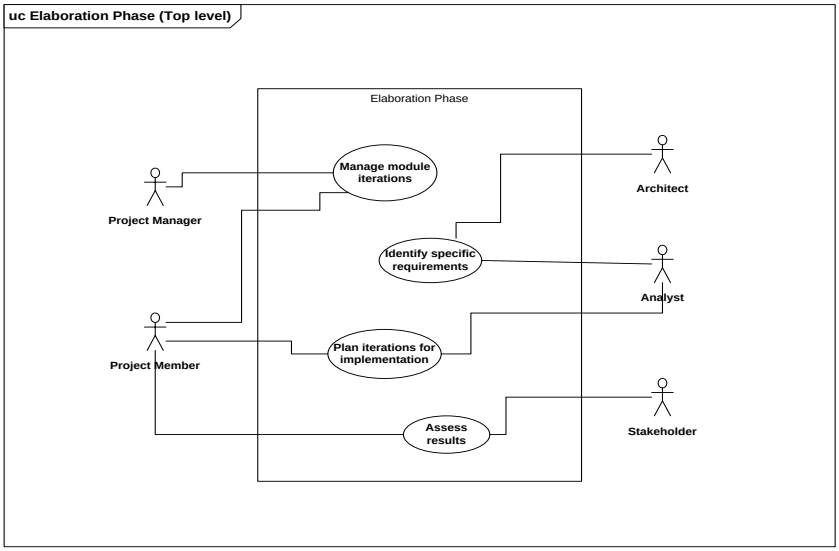

Fig. 5 CRM implementation Elaboration (use case diagram)

The *activity diagram* depicts the flow of tasks that should be completed in order to achieve the required results. Fig. 6 depicts the activities of the task Design the Solution.

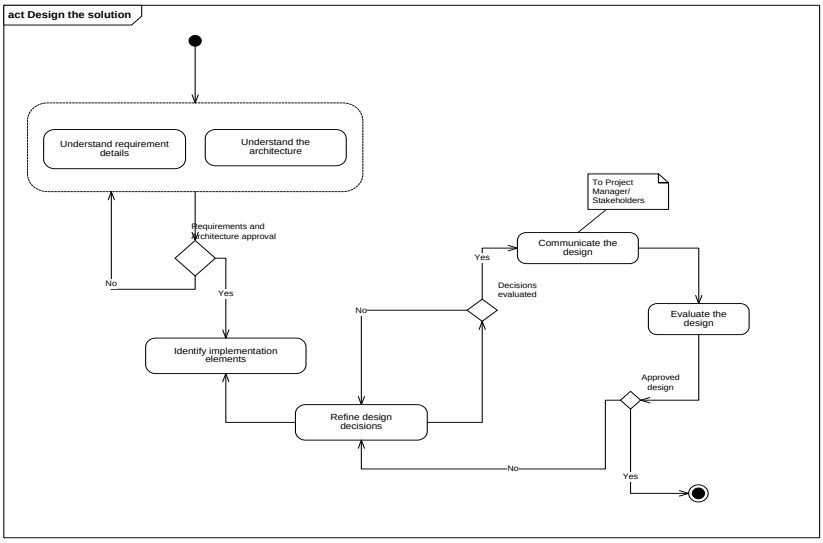

Fig. 6. CRM Design the solution (activity diagram)

The Requirements are represented according to the hierarchy and correspondence in the process, also, the relationships between them is shown. Activity diagrams depict

the process of one specific task, establishing the sequence of actions to follow and the conditions that need to be completed in order to finish the process.

4. - Conclusions

This work has explored the viability and use of SysML as an amenable tool for modeling CRM systems implementation services. We believe that SysML, when used effectively in a complete way, can show not the processes involved in an implementation method but also other method elements such as justiflying requierements, working roles, deliverables, etc., thus allowing a better comprehension and application of the resulting method in any project. Since the SysML language is a relatively new tool in academia, given its mostly practitioner origin, we thought it would be interesting to include it in our work in order to clarify some issues regarding the graphical definition of the method elements.

We have presented some examples of the use of SysML diagrams for the graphical representation of the Elaboration phase of our aimed implementation method for CRM systems. Because of space matters, we have only showed some of them, as well as a brief description of the process and elements they explain. Basic diagrams, such as block and activity diagrams are represented by modelling the main items of the Elaboration phase in each case, with the internal block diagram showing the activity and data flow inside.

We expect that the use of SysML for engineering services such as enterprise information systems implementation methods can contribute to a better development of their overall process, and also to simplify usual activities by reducing the work to be done as well as improve the quality and lower project risks. In the same way, the implementation services partially described on this article have been formalized with a language that shares elements of UML while adapting and extending it with additional diagrams. This may also facilitate to use SysML as the only notation for this purpose, instead of having to use several notations for the different components of an implementation service. From our initial experience, we think that other people involved in designing and formalizing software-related methods can also consider the use SysML for modelling software, hardware and service processes, since SysML integrates many elements from other modeling languages into a single notation.

Given the relative youth of SysML, there are not yet many tools supporting it; we have developed the diagrams showed in this paper with a Visio extension for SysML called *Stencil*, a very useful but basic tool for designing these kinds of graphics.

References

Boehm, B. Turner, R. (2003), "Using risk to balance agile and plan-driven methods", IEEE Computer Magazine, Vol. 36 Issue 6 pp. 57-66.

Bose, R. (2002) Customer Relationship Management: key components for IT success, *Industrial Management & Data Systems*, 102, 2, 89-97.

Bibiano Luis H; Caldelas-Lopez A; Mayol E. Pastor J.A.(2007) "Comparative Analysis Of CRM and SCM Systems Implementation Approaches" Online Proceedings of the European and Mediterranean Conference on Information Systems (EMCIS) 2007, 23-24 June; Valencia, Spain.

Bibiano, Luis H; Pastor, J.A; Mayol, E. (2009) "Organizational Issues in CRM System Implementation Projects". Proceedings of the IADIS International Conference on Information Systems 2009. 25-27 February, 2009. Barcelona.

Bibiano, Luis H; Pastor, J.A (2006) "Towards a definition of a CRM system lifecycle". Proceedings of the European and Mediterranean Conference on Information Systems (EMCIS) 2006, 5-7 July, Alicante, Spain.

Bull, C. (2003) "Strategic issues in customer relationship management (CRM) implementation", *Business Process Management Journal*, 9, 5.

Davenport, T.H. and Short, J.E. (1990), ªThe new industrial engineering: information technology and business process designº, *Sloan Management Review*, Vol. 31 No. 4, pp. 11-27.

Davenport, T.H. (2001) Putting the enterprise into the enterprise system, *Harvard Business Review*, July/August, 121–131.

Gartner Research (2003) "The CRM Reality and Hype Cycle".

Gefen, D. and Ridings, C.M. (2002) Implementation team Responsiveness and user evaluation of Customer Relationship Management, Journal of Management Information Systems, 19, 1, 47-69.

Hause, M. (2006). The SysML Modelling Language. Fifteenth European Systems Engineering Conference, September 2006.

Parvatiyar, A., Sheth, J.N. 2002. 'Customer Relationship Management: emerging practice, process and discipline'. *Journal of Economic and Social Research*, 3(2).