Negotiating Requirements for COTS-based Systems

Carina Alves, Anthony Finkelstein

Department of Computer Science, University College London, UK
{c.alves, a.finkelstein@cs.ucl.ac.uk}

Abstract. Selecting COTS products is a process that inherently involves tradeoffs. In this position paper, we highlight the challenges of the requirements process for COTS-based system where the negotiation of requirements plays a critical task. In particular, it is necessary to perform a careful balancing between stakeholders requirements and COTS features. We analyse the processes of COTS matching and balancing using a goal-driven approach. **Keywords**: COTS-based systems, negotiation process, goal modelling.

1. Introduction

COTS-based system development brings fundamental changes in how organizations do their work [3]. In particular, the requirements engineering process for such systems is affected by problems that are very different from those of traditional systems. COTS products are designed to meet the needs of a marketplace instead of satisfying the requirements of a particular organization and it is not ensured that available packages will meet all stated requirements. Therefore, during the evaluation of products it is necessary to perform an extensive process of requirements prioritisation and negotiation [13].

It is widely accepted that COTS procurement must be an interleaved process with requirements specification [3],[6],[9] in which the success of COTS-based systems depends on the effective evaluation of products. Selection is a time consuming activity, where a considerable amount of time is necessary to search and screen all potential COTS candidates. However, in practice many selection decisions are based on subjective judgements, such as current partnerships, commercial profits, and successful vendor marketing.

The evaluation process demands some form of inexact matching between products features and requirements specification where conflicts are identified. Due to the conflicting nature of this process, a continuous process of negotiation and tradeoffs should be performed. It is unrealistic to expect that selecting a suitable package is the main aspect of developing systems based in COTS [12], it is also necessary to take into account how the product will affect the organization business process and how the new releases will fit customers requirements. Therefore there is a critical need to balance the requirements of the organization against the features of the package. Our work aims to develop a better understanding of how this balancing should be carried out in order to support the COTS selection as well as analyse the impact of decisions over the system life cycle. This paper is organized as follows. Section 2 presents the main motivations of this research. Section 3 describes key issues of requirements modeling for COTS-based systems. In section 4 we conclude this work and explore some future work directions.

2. Motivation

In traditional systems development, the requirements engineering (RE) activity basically consists of eliciting stakeholders needs, refining the acquired goals into non-conflicting requirements statements, and finally validating these requirements with stakeholders. The main goal of the requirements engineer is to ensure that the requirements specification meets stakeholders' desires and it represents a concise and clear description of the system to be developed. Broadly speaking, the specified requirements will be translated into software architecture and ultimately, implemented. Therefore, it is reasonable to assert that requirements play a controlling role in system development [14]. The RE process for COTS-based development is affected by different problems from those of traditional systems [2]. Following we point out some important issues:

Dilution of Control - COTS are designed to satisfy very general requirements where the vendor has full control over the product releases and upgrades. Then, customers are put into unexpected situations over which they have no control.

Continuous Requirements Process - In COTS-based systems, requirements are extremely volatile mainly because of rapid changes in the COTS marketplace. Therefore, customers have to accept new releases with features that can be either unwanted or conflicting with stated requirements.

Requirements Flexibility - Requirements for COTS-based systems should not be so strict that either exclude the use of COTS or that require large product modification in order to satisfy very specific requirements [3].

The general motivation of our research is getting a deeper insight into the requirements process for COTS-based development. In this position paper we discuss the processes of matching and consequent balancing between requirements and COTS. In fact, a wide range of conflicts can arise during the matching between customers requirements and COTS features, ranging from simple misfitting of desirable features to severe problems of product integration into the organization.

3. Modelling Requirements for COTS-Based Systems

Goals have been recognized as a leading concept in the RE process [1],[4],[5],[8]. In this way, we follow a goal-oriented approach for requirements specification. This work is also influenced by the CORA approach for conflict resolution [11]. A bookstore COTS system selection is used as example to explain the presented concepts.

3.1 Goal Driven Modelling

The evaluation of COTS products starts with the acquisition of customer goals. From these initial goals, possible candidates are identified in the marketplace, where new goals may be recognized in features making this process highly iterative. Fig 1 presents part of the goal specification for the bookstore system using AND/OR refinement.

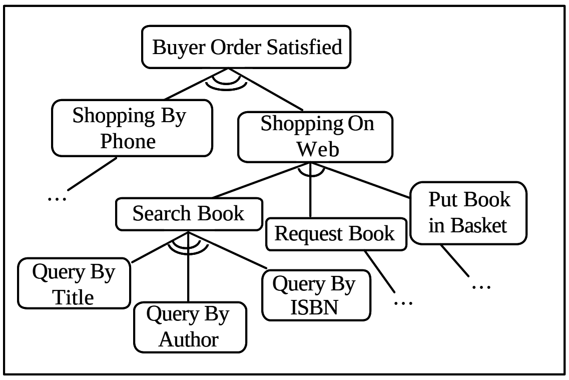

Fig. 1. Goal Refinement for Bookstore System

We have identified two categories of goals: peripheral and core. The former are goals that help to distinguish between products (i.e. goals not supported by all products) while the latter are goals that are provided by most available products then they are not very helpful for the decision process. During the evaluation, core goals are discarded while peripheral goals are stressed [7]. For example, the Goal (Query BookByTitle) is supported by both packages (example of core goal) while the Goal (KeepBuyerHistory) is supported only by product B (peripheral goal). Thus, this goal can be a decisive criterion and should be investigated in order to support the decision of selecting one product instead of others. We propose two goal attributes, desirability and modifiability that are formally specified bellow.

Desirability - the importance of a goal described in the customer specification to be satisfied by a particular feature provided by the package.
FormalDef - ∀ goal ∈ CustomerSpecs, ∀ feature ∈ PackageSpecs, Matching (goal,feature) ⇒ Verify (goal.Importance)

Modifiability - the capacity to restructure a goal definition when a conflict arises between a specified goal and any feature provided by the package.
FormalDef - ∀ goal ∈ CustomerSpecs, ∀ feature ∈ PackageSpecs, Matching (goal, feature) ∧ Conflict (goal, feature) ⇒ Modify (goal.Definition)

Desirability is concerned with the priority of goals; here one possibility is assigning numbers that represent the relative importance of goals as applied in multi criteria decision-making methods [9]. In particular, the number of hard goals should be minimised because even if any product fits well with a complex set of goals, the volatility of marketplace practically imposes that the fit will be short-lived [14]. Modifiability is a means of removing conflicts by specifying goals in more specific, more general or even in different ways. For example, consider the conflict between MaintainBuyerHistory and MaintainBuyerPrivacy (i.e. negative interdependency), where the former goal has *medium* desirability and the latter has *very high* desirability. In this case, the modifiability of MaintainBuyerHistory should be high in order to support the tradeoff resolution. In other words, to support the MaintainBuyerPrivacy goal satisfaction, there should be alternatives to restructure the MaintainBuyerHistory goal or to change this package feature.

3.2 Matching Goals and COTS

The matching process can be characterised as a negotiation problem between specifications, where COTS features should be able to satisfy goals within an environment – the environment specifies available resources and domain constraints. The evaluation of COTS demands some inexact matching with customer goals. For example, there may be goals not satisfied by any available package, goals satisfied by some joint packages, goals partially satisfied, features of package not initially requested but that can be helpful, features irrelevant or even unwanted. Moreover, there are some cases where goals cannot be entirely satisfied without considerable product adaptation and other cases where these goals must be compromised. An additional complication is that both goals and package specifications might have incompleteness and inconsistencies. In short, it is necessary to perform a complex negotiation process and to reach commitments between the involved parties.

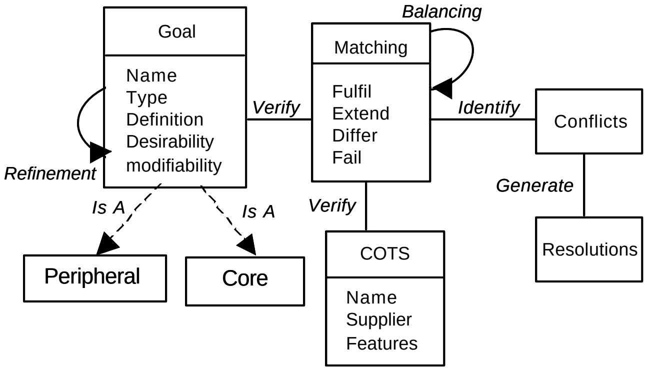

Fig. 3. Overview of the Matching Process

To support the matching process we have defined a taxonomy of matching pattern, some patterns are proposed:

Fulfil – feature exactly satisfies a customer goal

Extend – product provides an extra feature that was not requested by customer

Differ – product provides feature that partially satisfies a goal

Fail – product do not provide a feature that was requested by customer goal

Note that Fulfil pattern does not originate conflicts what means that Fulfil Features contribute positively for the satisfaction of customers goals. On the other hand, all other patterns might be a source of conflict and need to be explored. The Differ pattern can introduce conflicts as in this case the product has a feature that partially meets a particular goal but in which the feature differ in certain details. For example, product A partially meets the goal BookPriceDisplayed (i.e. Differ) but instead of displaying the price when the book is found as stated in the goal specification, the product displays the price only when the buyer requests the book. Consider as another example the following matching between goals and COTS features. As illustrated in Fig. 1, the Goal (SearchBook) is refined into the Subgoals (QueryByTitle), (QueryByAuthor) and (QueryByISBN). Product A supports the first two subgoals (i.e. Fulfil) but it also allows (QueryByPublisher) as an Extend Feature. On the other hand, Product B only Fulfil the Subgoal (QueryByTitle) and Fail the two others. We argue the analysis of matching patterns should support the identification of conflicting issues as sketched in Fig. 3.

3.3 Balancing Goals and COTS

The balancing between goals and COTS features is an important step of the decision making process. Assuming that conflicts are resolved through the generation of alternatives, a risk analysis should be performed as guidance in order to reach a feasible resolution (i.e. the solution that minimise potential conflicts and maximise goals satisfaction). In order to tackle the conflict problem it is necessary to understand the nature of conflict, analyse the causes of conflict, and explore the potential resolutions. We believe that an effective strategy for resolution generation should be a domain-independent one, where new conflict resolutions can be derived from previous resolutions of similar cases

4. Conclusions and Future Work

This position paper is a preliminary examination into the problem of matching between customers needs and package features. We argue that resolving conflicts that arise from the matching process is a critical issue for developing successful COTS-based systems. Without negotiation strategies, customers often focus on accepting a ready COTS solution (which will hardly meet their needs), rather than exploring new alternative solutions sharing mutual commitments. For example, customers may change their business practices in order to fit the product; products will probably need modifications that can range from simple customisations to large adaptations. As future work we need to investigate how to identify and characterise the various types of conflict that can arise from mismatches. Another major issue that needs to be addressed is the generation of negotiation strategies. Finally, once the approach will be fully defined, we need to empirically validate it in an industrial setting.

References

[1] Anton, A. Cracken, W. Potts, C. Goal Decomposition and Scenario Analysis in Business Process Reengineering. Proceedings of CAISE'94, 1994.

[2] Carney, D. COTS Evaluation in the Real World. SEI Interactive, Carnegie Mellon University, December 1998.

[3] Carney, D. Requirements and COTS-Based Systems: A Thorny Question Indeed. SEI Interactive, Carnegie Mellon University, June 1999.

[4] Chung, L. Nixon, B. Yu, E. and Mylopoulos, J. Non-Functional Requirements in Software Engineering. Kluwer Academic Publisher, 2000.

[5] Dardenne, A. Lamsweerde, A. Fickas, S. Goal-directed Requirements Acquisition. Science of Computer Programming. Vol 20 North Holland, pp 3-50, 1993.

[6] Deifel, B. Requirements Engineering for Complex COTS. 4th International Workshop on Requirements Engineering: Foundation for Software Quality. Pisa, Italy 1998.

[7] Finkelstein, A. Spanoudakis, G. Ryan, M. Software Package Requirements and Procurement. Proceedings of the 8th International Workshop on Software Specification and Design, 1996.

[8] Lamsweerde, A. Goal-Oriented Requirements Engineering: A Guided Tour. Invited mini-tutorial paper 5th IEEE International Symposium on Requirements Engineering. Toronto, Canada. August 2001.

[9] Ncube, C. Maiden, N. A. M PORE: Procurement-Oriented Requirements Engineering Method for the Component-Based Systems Engineering Development Paradigm. International Workshop on Component-Based Software Engineering, May 1999.

[10] Robinson, W. Negotiation Behavior During Requirements Specification, 12th Conference on Software Engineering, 1990.

[11] Robinson, W.N., Volkov, S., Conflict-Oriented Requirements Restructuring, GSU CIS Working Paper 99-5, Georgia State University. April 1999.

[12] Rolland, C. A Goal driven Approach to Modelling COTS Acquisition Impacts. IFIP International Conference on Enterprise Management and Resource Planning Systems Method, Tools and Architectures (EMPRS). Pisa, Italy. November 1999.

[13] Sivzattian, S. Nuseibeh, B. Linking the Selection of requirements to Market Value: A Portfolio-Based Approach. 7th International Workshop on Requirements Engineering: Foundation for Software Quality. Interlaken, Switzerland, June 2001.

[14] Wallnau, K. Hissam, S. Seacord, R. Building Systems from Commercial Components. SEI Series in Software Engineering. Addison Wesley, 2002.