

# Using software process modeling to analyze the COTS based development process

Reidar Conradi<sup>1</sup>, Letizia Jaccheri<sup>1</sup>, Marco Torchiano<sup>2</sup>

<sup>1</sup> Department of Computer and Information Science, Norwegian University of Science and Technology, Sem Sælands vei 7-9, 7491 Trondheim, Norway  
conradi@idi.ntnu.no, letizia@idi.ntnu.no

<sup>2</sup> Dipartimento di Automatica e Informatica, Politecnico di Torino, C.so Duca degli Abruzzi 24, 10129 Torino, Italy  
torchiano@polito.it

## Abstract

Several process guidelines for the development of COTS-based software systems have been described by informal notations and published in literature. We present here our attempt to model the COTS-based software process using a software process modeling language (E3). We provide models of existing COTS-based development processes, we discuss some of their limitations, and we give suggestions for improvement of these.

## 1 Introduction

As technology evolves, new processes for developing and maintaining software system continuously appear on the market. COTS (Commercial-Off-The-Shelf) based development is one example of a class of software processes which have been recently proposed and studied by the software engineering community [1] [2] [3] [4] [5]. The COTS-based development process involves both the vendor organization that produces, sells, or distributes COTS products, the integrator organization that acquires and assembles these products, and the customer organization that purchases and pays for the final software system.

One of the goals of the software process modeling discipline is process understanding and analysis by synthesis [6]. Our idea is that process modeling can be used as a tool to model existing COTS based development processes. In this short paper, we present two COTS-based development processes extracted from literature. We provide an abstract model of these by means of a Process Modeling Language and we use this process model as a basis for proposing some improvements.

## 2 Models of existing COTS-based development processes

The COTS-based development processes proposed in the literature are described by informal notations. Figure 1 and Figure 2 present two examples of COTS-based development processes.

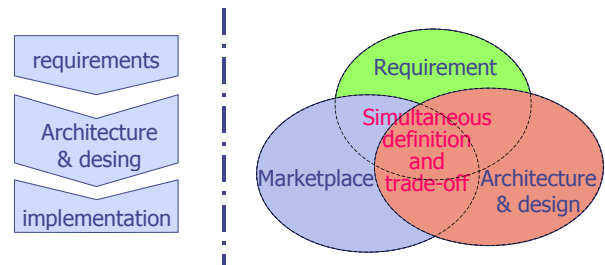


Figure 1: COTS-based development as proposed by [2].

Figure 1 conveys the idea that the COTS-based development is a continuous trade-off between several factors: the requirements for the CBS, the COTS products offered by the marketplace, and the architecture of the CBS. However, the relationships between the software development activities (requirement analysis, architecture and design, and implementation), their responsible roles, and their deliverables are left to the interpretation of the reader.

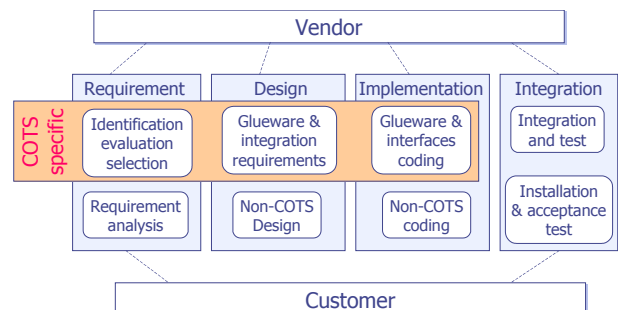
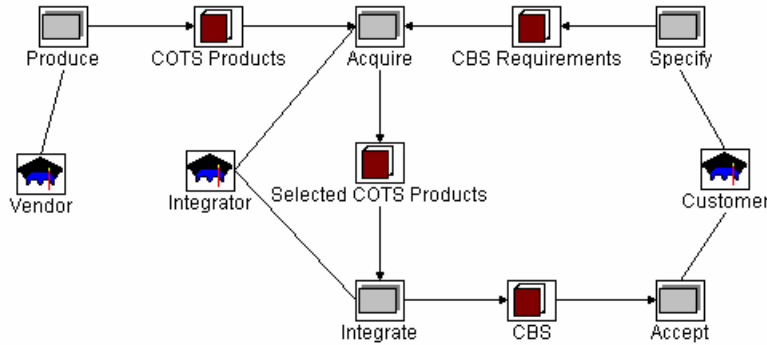


Figure 2: Simple life cycle of COTS-intensive systems as proposed by [1].

Figure 2 describes a COTS-based software development process. The process consists of four phases



**Figure 3: COTS-based development: a simplified view.**

(Requirement, Design, Implementation, and Integration), each of which is split into a COTS specific and a Non-Cots specific one. This picture identifies two roles: the COTS vendor organizations, and the customer organization. With respect to the process in Figure 1, the process in Figure 2, defines the need for a phase at the beginning of the development where identification, evaluation, and selection of COTS products take place.

In our project, we have formally described an archetypical COTS-based development process by means of a Process Modeling Language (PML). Both the process model itself and the modeling activity allow us to reason about the process to identify drawbacks and propose changes. As we have experience with the E3 PML [7] and its associated tool, we adopt this environment to describe processes. We also propose an extended COTS based development process in the form of a software process model. This process has been empirically validated in several occasions [8-10].

The process model we developed tries to integrate the aspects described by the different models found in the literature and those emerging from our experience. Here we give some fragments of the process models we have developed. For each fragment, hints for discussion are given.

Figure 3 presents a simplified process model of the general COTS-based development process. We use a subset of the E3 notation. Rectangles denote tasks. Books denote artifacts. Hats denote roles. Arrow between tasks and artifacts denote input and output relationships.

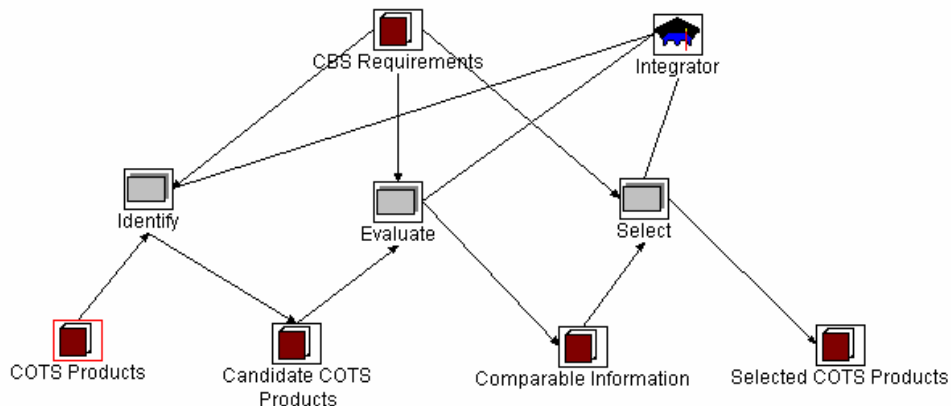
Connections between tasks and roles denote responsibility. The model is based on those found in the literature and consists of a limited set of concepts. This model defines the context to reasoning about COTS-based development. In our model, vendors produce and/or distribute COTS products and the integrator acquires one or more COTS products and builds them into a COTS-based system (CBS). Finally, the customer purchases the CBS.

This model fragment identifies three main roles. First, the *Vendor* is the organization that produces, sells, or distributes the COTS products. The Vendor is the source for both the product itself and its documentation. Often it provides some kind of support services which may be free or sold separately. Second, the *Integrator* is the organization that builds the COTS-based system assembling COTS products, possibly together with internally developed pieces of software (often called glueware) and components. Finally, the *Customer* is the organization that purchases and pays for the CBS. In this simplified view, requirements for the CBS come directly from the customer.

The proposed high-level model is in agreement with the one proposed in [1] about the actors that interact in the CBD process.

This is a general process model fragment whose main goal is to provide a context for the Acquire task, which is discussed below.

Figure 4 depicts the structure of the Acquire task. This is



**Figure 4: The structure of the Acquire task.**

also based on acquisition processes found in the literature (e.g. [11] [12]); in particular we find the same activities (i.e. identification, evaluation, and selection) described in the model presented in [1].

The two process models presented so far describe a picture of the current state of the art in the literature.

Based on our experience, both interacting with the industry and teaching to students, in the development of COTS –based systems we identified some drawbacks and several areas for improvement.

### 3 Suggestions for improvement of existing COTS-based development processes

By analyzing the acquisition process model, we observe at least two directions for improvement. First, the identification phase of the acquisition process must be extended to support COTS products discovery. Then, the qualitative data that is available about COTS products must be managed and acquired by the integrator organization. Finally, the relationship between the architecture of the final system and the selected COTS products must be managed.

The process model fragment presented in Figure 5 proposes an improved Acquire task. With respect to the process model in Figure 4 there is a new task, called Learn task. The learning phase produces a set of COTS products that can be used as a starting point for the identification of potentially useful products. In addition, a preliminary software architecture is defined which can be used to drive subsequent identification and evaluation tasks. We have proposed a process model for this Learn task. We will also provide a discussion around the process model proposed in Figure 5. This model in fact can be added by important details which express the trade-off between the CBS Requirements and the

Integration phase.

Acquisition approaches proposed in the literature are all attribute-based, intended to measure product suitability with a set of stated requirements. The relationship between COTS requirements and the CBS requirements depends on the system type. For turnkey systems the two requirements are essentially the same. In the case of integrated systems the transition between COTS and CBS requirements should be mediated by software architecture. In this work we choose to adopt the definition of software architecture provided in [13]: “The software architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components and the relationships between them”.

The architectural components represent placeholders for COTS products, and the architecture as a whole dictates product requirements. In order to design such architecture, it is important to know which components can be used. Otherwise we may end up with an unfeasible architecture. Therefore, the integrator must have a thorough understanding of candidate COTS products before he or she can make an appropriate selection. Such knowledge may come from different sources, depending on the integrator organization's experience with CBS domains and application areas. Experienced developers may possess all the knowledge required to design a feasible architecture. Whenever such experience is not available, however, we need a means to procure the required knowledge.

### 4 Conclusions

In summary, several COTS-based development processes have been proposed in the literature. These are different from the traditional water-fall like software processes

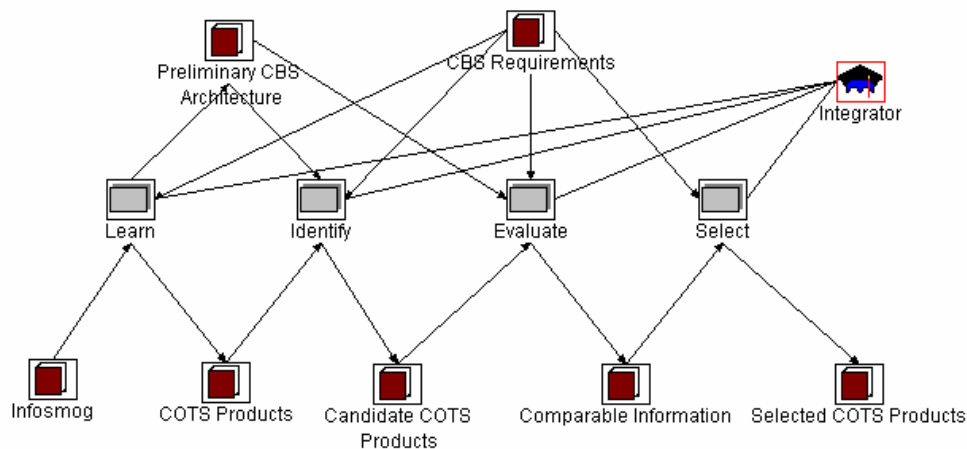


Figure 5: A proposal for an enhanced Acquire task.

since they are characterized by simultaneous definition of requirements, architecture and design, and trade-off between requirements and available COTS on the market place. In our work we have studied the existing COTS-based processes and provide software process models. These models facilitate the discussion about COTS-based development and give directions for improvement. In this position paper, we have provided an example of a general acquisition process and its improvement. The message is that although COTS-based development processes are neither linear nor simple, the application of process modeling techniques leads to models which can be analyzed, compared, discussed upon, and improved.

## 5 References

- [1] M. Morisio, C. Seaman, A. Parra, V. Basili, S. Condon, and S. Kraft, "Investigating and Improving a COTS-Based Software Development Process," presented at 22nd International Conference on Software Engineering, Limerick, Ireland, 2000.
- [2] L. Brownsword, T. Oberndorf, and C. A. Sledge, "Developing New Processes for COTS-Based Systems," *IEEE Software*, vol. 17, pp. 48-55, 2000.
- [3] D. Carney and K. Wallnau, "A basis for Evaluation of Commercial Software," *Information and Software Technology*, vol. 40, pp. 851-860, 1998.
- [4] P. Lawlis, K. Mark, D. Thomas, and T. Courtheyn, "A Formal Process for Evaluating COTS Software Products," *IEEE Computer*, vol. 34, pp. 58-63, 2001.
- [5] K. Wallnau and A. Brown, "A Framework for Evaluating Software Technology," *IEEE Software*, vol. 13, pp. 39-49, 1996.
- [6] J. Derniame, B. Ali Kaba, and D. Wastell, *Software Process: Principles, Methodology and Technology*, vol. 1500: Springer-Verlag., 1999.
- [7] L. Jaccheri, P. Lago, and G. P. Picco, "Eliciting Software Process Models with the E3 Language," *ACM Transactions on Software Engineering and Methodology*, vol. 7, pp. 368-410, 1988.
- [8] L. Jaccheri and M. Torchiano, "Classifying COTS products," presented at 7th European Conference on Software Quality (ECSQ 2002), Helsinki, Finland, 2002.
- [9] M. Torchiano, L. Jaccheri, C.-F. Sørensen, and A. I. Wang, "COTS Products Characterization," presented at Conference on Software Engineering and Knowledge Engineering (SEKE'02), Ischia, Italy, 2002.
- [10] M. Torchiano and L. Jaccheri, "Assessment of Reusable COTS Attributes," presented at 2nd Int. Conference on COTS Based Software Systems, Ottawa, Canada, 2003.
- [11] N. Maiden and C. Ncube, "Acquiring COTS Software Selection Requirements," *IEEE Software*, vol. 15, pp. 46-56, 1998.
- [12] M. A. Ochs, D. Pfahl, G. Chrobok-Diening, and B. Nothhelfer-Kolb, "A COTS Acquisition Process: Definition and Application Experience," presented at 11th ESCOM Conference, Shaker, Maastricht, 2000.
- [13] L. Bass, Clements, P., Kazman, R., *Software Architecture in Practice*: Addison-Wesley, 1998.