Barriers to Disseminating Off-The-Shelf Based Development Theories to IT Industry

Jingyue Li[1], Reidar Conradi[1,2], Odd Petter N. Slyngstad[1]
[1]Dept. of Computer and Info. Sci.
Norwegian Univ. of Sci. and Tech.
NO-7491 Trondheim, Norway
[2]Simula Research Laboratory, P.O.BOX 134, NO-1325 Lysaker, Norway
{jingyue, conradi, oslyngst}@idi.ntnu.no

Christian Bunse[3] Umair Khan[3]
[3]Fraunhofer IESE, Sauerwiesen 6,
D-67661 Kaiserslautern, Germany
{Christian.Bunse, khan}@iese.fraunhofer.de

Maurizio Morisio[4] Marco Torchiano[4]
[4]Dip. Automatica e Informatica, Politecnico di Torino
Corso Duca degli Abruzzi, 24,
I-10129 Torino, Italy
{maurizio.morisio, marco.torchiano}@polito.it

ABSTRACT
In this position paper, we have reported results of an industrial seminar. The seminar was intended to show our findings in an international survey, conducted in Norway, Italy and Germany, on off-the-shelf component-based development. Discussion in the second section of the seminar revealed several obstacles of popularizing the OTS based development theories into IT industry.

Keywords
COTS component, OSS component, Empirical Study

1. INTRODUCTION
Software developers are using an increasing amount of COTS (Commercial-Off-The-Shelf) and OSS (Open Source Software) components in their projects. However, using such external components introduces many risks [2, 5, 10]. Before project managers decide to acquire an external component, instead of building it in-house, they must identify and mitigate possible risks. Researchers and practitioners have proposed many new COTS selection and development processes [1, 14,19].

As most theories were based on local case studies, it is necessary to evaluate their conclusions on a larger sample. We have performed an international survey on process improvement and risk management in COTS and OSS component based development in Norway, Germany and Italy [12, 13]. After we analyzed data, we hold an industrial seminar to report our preliminary results to the participants of this study and some interested project managers. An hour discussion revealed many new concerns from an industrial viewpoint.

2. BACKGROUND
Our survey is the second phase of a systematic study on process improvement and risk management in OTS based development. We started from a pre-study, focusing on software process improvement in COTS based development, and being performed with structured interviews of 16 COTS projects in Norwegian IT companies [11]. The following main survey extended the pre-study in two dimensions. First, it included OSS components because they represent an alternative to COTS components. Second, this study included samples from Norway, Italy and Germany. In addition, the sample was selected randomly instead of on convenience as in the pre-study.

The survey was designed to investigate six main research questions:

- RQ1: What are the main motivations of using OTS components instead of building them in-house?
- RQ2: What are the real common OTS-based process models, and what are the possible variances?
- RQ3: What is the OTS component selection process used in practice?
- RQ4: What risks have been identified and what are their relationships with the project contexts?
- RQ5: Which risk management methods have been performed and with what the results?
- RQ6: What are the differences between COTS and OSS components?

The unit of study is a completed software development project. The projects were selected based on two criteria:

- The project should use one or more OTS components
- The project should be a finished project, possibly with maintenance, and possibly with several releases.

In the data collection process, we promised our participants that we would report the result of the study in the form of a seminar. After we have finished the data collection in Norway (47 projects) and gathered some data from Germany and Italy (23 from Germany and 20 from Italy), we held a seminar at Oslo in Feb. 2005 as we promised to the Norwegian participants.

The intentions of the seminar include two parts:

- First, we would like to report our results by giving guidelines on how to improve processes and manage risks in OTS based development.
- As most data of this study show only the state-of-the-practice, we do not know possible cause-effect relationships

of our conclusions. We therefore had one hour discussion as the second part of the seminar to get feedback from the participants.

23 Norwegian industrial persons showed up in the seminar. More than 40% of them came from large-sized IT companies (more than 100 employees). 40% came from a medium-sized (with 20-99 employees) and 20% came from small-sized companies (less than 20 employees). We first showed our survey results in one hour presentation. On-going our talk, participants were able to interrupt and ask questions. In the second part, we hypothesized some questions based on the survey result. We also listed our assumed explanations on these questions. Industrial participants gave their opinions to support or reject our explanations.

3. RESULTS

Concerns and comments from industrial colleagues gave us valuable ideas on what we need to do in order to disseminate OTS based development theories into industry. We have summarized industrial concerns and comments into five barriers.

3.1 The barrier to disseminating the overall formal OTS-based development process

The pre-study showed that most projects decided their main development process before they started to think about using OTS components [11]. The result from the survey follows the same trend. In the discussion section, we asked why they did not use the proposed OTS-based development processes as in [3,14]. Our assumed explanations were:

1. Not necessary, as OTS is a minor part of the system
2. It is difficult to change (company level)

Industrial answers: Industrial colleagues agreed with our second explanation. They said people would not like to change the way they work. They think it is either too expensive or too difficult. They argued that there were few empirical studies to investigate the cost and benefit of the proposed COTS-based development processes. It is therefore difficult to convince industrial developers to use them. They do not want to change their development processes dramatically, being a painful procedure, without enough confidence.

Relevant studies: Some previous studies have analyzed the barriers of using COTS information technology based on lessons learned in small and medium sized companies (SMEs) [18]. The conclusions are:

- SMEs do not formally define their processes and tend to use tribal knowledge.
- SMEs use flexible processes as a competitive advantage and take pride in that flexibility. Changing this approach is difficult, because many SMEs equate process flexibility with having undocumented processes.
- The culture within SMEs is one of informal teamwork and therefore is resistant to formalism or change.

Our insight: In our study, we excluded companies with less than 5 employees because most such companies do not have a formally defined process. Most companies in our study have one de-facto process, such as RUP or CMMI. The reason for not using a formal COTS process is not because they do not have any formal development process. However, before adopting their process into a formal process, such as EPIC [1], the industrial practitioners will need evidences that the formal process is more cost-effective. It is also necessary for them to know how to downsize a formal process according to different project contexts, such as time-to-market pressure, quality requirements and so on.

Barrier one: Without sufficient empirical evidence on OTS based development processes, industrial project managers would not like to change their de-facto development process.

3.2 The barrier to disseminating the formal OTS selection process

Although many formal OTS selection processes have been proposed [4, 9], results of this study and some other studies [20] showed that they were seldom used in practice. In the discussion section, we asked why they did not use these formal selection processes. Our assumed explanations were:

1. Don't know the formal selection process
2. Not necessary
3. It was not easy to use

Industrial answers: Industrial participants agreed with our second explanation. They argued that OTS component are not the key parts of the system in some cases. They just want to select one as fast as they can. These formal selection processes are not cost-effective for them. In addition, they argued that there were few empirical studies on when they should use the formal process instead of downloading a demonstration and trying it out, as the process discovered in study [11]. They were few studies showing that a formal process actually gave better results than informal processes.

Relevant studies: The balance between COTS uncertainty risks and risks resulting from project delay has been investigated by an empirical study [16]. They concluded *COTS assessment appears qualitatively intuitive for any given aspect. However there are often many factors that must be considered simultaneously. The impact of all considerations may be quite complex and counter intuitive.*

Our insights: Most formal selection processes were proposed without evaluations on their pre-conditions and their cost-benefit. For example, the selection process for fine-grained small OTS component (the object of this study) would be different with selection process for large OTS suites. The large OTS suites usually have more features than required from the customer. How to evaluate these extra features have not been investigated in current formal selection processes [15]. In addition, most selection processes assumed that the general software architecture was ready before OTS selection. However, our studies discovered that the OTS component can be selected in the pre-study phase, requirement phase, general design phase, detailed phase, and coding phase [11]. That is, how to customize the formal selection process according to little information needs future studies.

Barrier two: Little advice on how to customize a formal process according to project context and OTS itself prevents the use of a formal selection process.

3.3 The barrier to sharing OTS based development knowledge inside company

Our pre-study and some other studies [14] showed that a new role, an OTS knowledge keeper, is necessary in COTS-based development, Result of this survey showed that projects with a dedicate person keeping fresh of the knowledge of OTS component had fewer problems (see problems we listed in [12]) than projects without it. The results also show that the more project members had general experience with OTS-based development, the fewer problems they had in their projects. In the discussion section, we asked what other knowledge is necessary, in addition to the technical insights in concrete OTS components.

Industrial answers: Industrial participants argued that general risk management and cost estimation experience must be shared. However, they had no good idea on how to do it. Moreover, they were interested in how much effort the knowledge keeper needs to spend on OTS component based development and what kind of knowledge needs to be kept.

Relevant studies: In study [6], the authors concluded that a pre-fabricated, reusable set of generic requirements, based upon the practical analysis in business environments of application and COTS middleware components characteristics helped to ensure the success of COTS middleware selection.

Our insight: Knowledge accumulation and transfer inside a software company and software projects have been investigated in other software development fields. However, what knowledge should be documented and shared in OTS-based development has seldom been investigated. It is therefore difficult for industrial practitioners to collect and share relevant OTS based development knowledge.

Barrier three: Few studies on knowledge accumulation and transfer delayed the dissemination of the OTS based development experience.

3.4 The barrier to using OSS components

Our study compared the differences between projects using only COTS component with project using only OSS components. Results show that OSS users have serious problems in getting the support reputation information on OSS component providers [13]. We asked the question: How to evaluate the support reputation of OSS community?

Industrial answers: The conclusion from industrial participants is that it is not possible to get this information and ensure the technical support from OSS providers currently. They argued that most available information of OSS component support come from the previous experience, i.e. the previous response time, the bug fixing speed etc. Without a commercial contract, as with a COTS vendor, it is not possible to ensure the future response time and the quality of technical support because most technical support work is based on volunteers.

Relevant studies: OSS component have been used more popular in commercial projects. As discussed in [7], how to establish credible and mutually acceptable combinations of OSS delivery and procurement models therefore needs future research.

Our insight: Industrial practitioners inclined to use OSS components in their projects. However, concerns on incredible technical support from OSS blocked the process.

Barrier four: Without credible and mutually acceptable combinations of OSS components' technical support, it is difficult for an industrial developer to use OSS in practice.

3.5 The barrier to doing requirements (re)negotiation in OTS-based development

Our pre-study showed that customers' involvement on OTS component selection and decision making could be a good risk mitigation strategy. It gives integrators the leeway to (re) negotiate requirements when the limitations of an OTS component are found at the very end [11]. However, data of this study shows that this strategy has been seldom used in practice. We asked the question: why customer had seldom been involved in COTS component decision making?

Industrial answers: Industrial colleagues explained that their customers either had no technical background to make such decisions or cared only the final product without considering the technical details. So, the integrators had to take all the responsibilities on OTS component decision making. As a result, the requirements (re)negotiation because of OTS component's limitations is impossible in practice.

Relevant studies: Methods of requirements (re)negotiation have been proposed as an important part of requirements engineering. However, both our pre-study [11] and another study [17] found out the same phenomena. If the COTS candidates could not feasibly satisfy pre-defined system project objectives, constraints, and priorities, these project objectives, constraints, and priorities could not be changed or negotiated to accommodate the constraints imposed by the available COTS. Therefore, building the required functionalities in-house became the only viable option.

Our insight: As it is difficult to discover all OTS components' limitations at the very beginning, it is necessary for integrators to be able to (re) negotiate the requirements with the customer. How to involve customers in OTS component decision making therefore needs future studies.

Barrier five: Few case studies have observed successful requirements (re)negotiation in OTS based development. There are therefore few guidelines on when and how to involve customers in requirements (re)negotiation because of OTS limitations.

4. CONCLUSION AND FUTURE WORK

This paper has reported some industrial obstacles to conduct OTS based theory into practice. We believe that answers to these barriers will help to improve OTS-based development practice in industry. In the future, we plan to study on these barriers by case studies and personal interviews.

5. ACKNOWLEDGMENTS

This study was partially funded by the INCO (INcremental COmponent based development) project [8]. We thank the colleagues in these projects, and all the participants in the survey and seminar.

REFERENCES

[1] Albert, C. and Brownsword L. Evolutionary Process for Integrating COTS-Based System (EPIC): An Overview. SEI, Pittsburgh.

http://www.sei.cmu.edu/publications/documents/02.reports/02tr009.html. 2002.

[2] Boehm, B. W., Port, D., Yang, Y., and Bhuta, J. Not All CBS Are Created Equally COTS-intensive Project Types. In *Proc. of the 2nd International Conference on COTS-Based Software Systems (ICCBSS'03)* (Ottawa, Canada, February 2003), LNCS Vol. 2580, Springer-Verlag, 2003, 36-50.

[3] Brownsword, L., Oberndorf, T. and Sledge, C. Developing New Processes for COTS-Based Systems. *IEEE Software, 71, 4,* (July/August, 2000), 48-55.

[4] Dorda, S. C., Dean, J. C., Morris, E., and Oberndorf, P. A Process for COTS Software Product Evaluation. *In Proc. of the 1st Int. Conf. on COTS-Based Software Systems (ICCBSS'02),* (Orlando, FL, USA, Feb. 2002), LNCS Vol. 2255, Springer-Verlag, 2002, 176-187.

[5] Fitzgerald, B. A Critical Look at Open Source. *IEEE Computer, 37, 7,* (July 2004), 92-94.

[6] Gorton, I. and Liu, A. Streamlining the Acquisition Process for Large-Scale COTS Middleware Components. In *Proc. of the 1st Int. Conf. on COTS-Based Software Systems (ICCBSS'02),* (Orlando, FL, USA, Feb. 2002), LNCS Vol. 2255, Springer-Verlag, 2002, 122-131.

[7] Holck, J., Larsen, M. H., and Pedersen, M. K. Managerial and Technical Barriers to the Adoption of Open Source Software. In *Proc. of the 4th Int. Conf. on COTS-Based Software Systems (ICCBSS'05),* (Bibao, Spain Feb. 2005), LNCS Vol. 3412. Springer-Verlag, 2005, 289-300.

[8] INCO project description, http://www.ifi.uio.no/~isu/INCO, 2000

[9] Lawlis, P. K., Mark, K. E., Thomas, D. A., and Courtheyn, T. A Formal Processes for Evaluating COTS Software Products. *IEEE Computer, 34, 5,* (May 2001), 58-63.

[10] Lawton, G. Open Source Security: Opportunity or Oxymoron? *IEEE Computer, 35, 3,* (March 2002), 18-21.

[11] Li, J., Bjørnson, F. O., Conradi, R., and Kampenes, V. B. An Empirical Study of Variations in COTS-based Software Development Processes in Norwegian IT Industry. In *Proc. of the 10th IEEE Int. Metrics Symposium (Metrics'04),* (Chicago, USA, September 2004), IEEE Press, 2004, 72-83.

[12] Li, J., Conradi, R., Slyngstad, O. P. N., Torchiano, M., Morisio, M., and Bunse, C. A State-of-Practice Survey on Risk Management in Off-the-Shelf Component-Based Development. In *Proc. of the 4th Int. Conf. on COTS-based Software System (ICCBSS'05),* (Bibao, Spain Feb. 2005), LNCS Vol. 3412 Springer-Verlag, 2005, 278-288.

[13] Li, J., Conradi, R., Slyngstad, O. P. N., Bunse, C., Khan, M., Torchiano, M., and Morisio, M. An Empirical Study on Off-the-Shelf Component Usage in Industrial Projects. Forthcoming in *the 6th Int. Conf. on Product Focused Software Process Improvement(PROFES 2005)* (Oulu, Finland, June, 2005)

[14] Morisio, M., Seaman, C.B., Parra, A. T., Basili, V. R., Kraft, S. E., and Condon, S. E. Investigating and Improving a COTS-Based Software Development Process. In *Proc. of the 22nd Int. Conf. on Software Engineering (ICSE 2000).* (Limerick, Ireland, June, 2000). ACM Press, New York, NY, 2000, 31-40.

[15] Perrone, V. A Wish List for Requirements Engineering for COTS-Based Information Systems. In *Proc. of the 3rd Int. Conf. on COTS-Based Software Systems (ICCBSS'04)* (Redondo Beach, CA, USA, Feb. 2004), LNCS Vol. 2959, Springer-Verlage, 2004, 146-158.

[16] Port, D. and Chen, S. Assessing COTS Assessment: How Much Is Enough? In *Proc. of the 3rd Int. Conf. on COTS-Based Software Systems (ICCBSS'04)* (Redondo Beach, CA, USA, Feb. 2004), LNCS Vol. 2959, Springer-Verlage, Berlin Heidelberg New York (2004), 183-198.

[17] Port, D. and Chen, S.,: Empirical Analysis of COTS Activity Effort Sequences. *Proc. of the 3rd Int. Conf. on COTS-Based Software Systems (ICCBSS'04)* (Redondo Beach, CA, USA, Feb. 2004), LNCS Vol. 2959, Springer-Verlage, 2004, 169-182.

[18] Robert, J., Buhman, C., Garcia, S., Allinder, D., Bringing COTS Information Technology into Small Manufacturing Enterprises. In *Proc. of the 2nd Int. Conf. on COTS-Based Software Systems (ICCBSS'03),* (Ottawa Canada, Feb. 2003), LNCS Vol. 2580. Springer-Verlag, 2003, 187-195.

[19] SEI (Software Engineering Institute). COTS-Based Initiative Description. http://www.sei.cmu.edu/cbs/cbs_description.html. 2004

[20] Torchiano M. and Morisio M., "Overlooked Facts on COTS-based Development", *IEEE Software, 21, 2,* (March/April 2004), 88-93.