Process Technology Implications of Procurement Processes: Some Initial Observations

Ernst Ellmer, Wolfgang Emmerich and Anthony Finkelstein

Dept. of Computer Science, University College London
Gower Street, London WC1E 6BT, UK
{E.Ellmer|W.Emmerich|A.Finkelstein}@cs.ucl.ac.uk

Abstract. We report on a study of procurement processes in a large organization. The purpose of the study was to identify problems in the organization's procurement processes and to suggest improvement actions. Procurement processes determine the characteristics of software processes. Procurement processes are themselves complex and amenable to process technology. Cost and scheduling benefits can be realised if procurement and contracting organizations integrate their respective processes. We conclude by highlighting the implications of procurement processes for the process research agenda.

Position Paper

1 Introduction

Software development processes are always embedded in larger scale processes, be they procurement, system engineering or business processes. Many of the features of software processes are determined by them. Examples include deadlines, budgets, base technology.

The software process research community has focussed on models, languages and tools to describe software processes. Little attention has been paid to the requirements for process technology that arise from the broader processes they are embedded in, as to how process technology can be applied used in these processes.

[1] reports on procurement from a requirements engineering point of view. In this paper, we focus on procurement processes. We report on some lessons that we have learned from a detailed procurement process study in a large organization[1]. While the organization used to build systems in the past it is now, in common with many organizations, only procuring and operating systems. In case of the organization we studied, the budget for a single project may be as much as £500 million. The procurement processes of large organizations are themselves extremely complex.

[1] Unfortunately, we cannot disclose the organization's identity in this paper.

The main points of this paper are:

- For defining system and software development processes the environment in which these are embedded needs to be understood and considered.
- Procurement organisations benefit if they become more aware of the processes they engage in.
- An effective integration is needed between the processes run by procurer and contractor.

This paper is structured as follows. In Section 2, we briefly present the study we performed, state the main characteristics of the organization and characterize the processes we examined. Section 3 identifies key problem areas in the procurement process of the organization. In Section 4 we derive some implications of procurement processes for software process technology.

2 Process Study

2.1 Approach

The approach we adopted for this project comprises four main steps and can briefly be characterised as follows. In the first step, we performed a large number of structured interviews with employees of the organization. We achieved good coverage of the different roles involved in systems engineering; we talked to people from different departments, such as research and development, contracts and purchasing, infrastructure, engineering and operational units; we also interviewed people from different hierarchical levels including project managers, requirements managers, requirements engineers and testers. The interviews were structured according to an *aide-memoire* that we developed for this purpose.

We minuted interviews and used the minutes to distill key process problems. We then formulated a set of suggestions to address the problems we identified. Finally, we outlined a programme of improvement actions for putting our suggestions into practice.

2.2 The Organization

The organization under consideration is very large with approximately six thousand employees. At the moment, the organization runs approximately 20 procurement projects concerned with the development of large software-intensive systems. These systems may be safety critical or contain substantial safety-critical components. The organization is geographically dispersed across numerous locations in the UK.

A main characteristic, very important for our study, is that the organization is very heterogeneous. It has departments that can respond to invitations to tender in the same way as external contractors. Departments may find themselves competing against each other. The organization has a strong project culture with relatively little corporate support for projects. Project teams are in charge of acquiring the necessary expertise for project performance and do not, in general, share expertise and knowledge. As the political and commercial environment develops, the organization is shifting away from

procuring systems towards procuring services. This development increases the heterogeneity of the organization, strengthening the tendency for internal competition and reinforcing the project culture.

2.3 Procurement Processes

The process for procuring a large and complex software-intensive system is long and complex. It involves numerous internal and external participants, including internal organizational units located in the UK as well as contractors and co-operating organizations located in elsewhere across Europe.

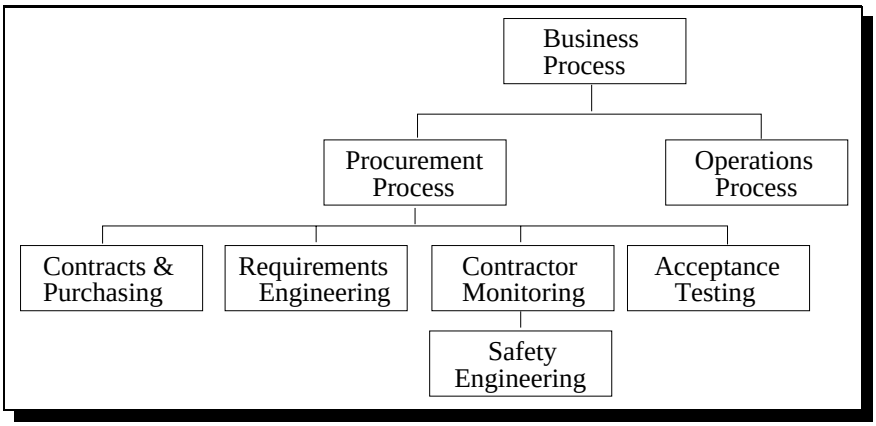

Fig. 1. Overview of Processes involved in Procurement

As shown in Figure 1, the *procurement process* is embedded in the overall *business process* of the organization. *Operations processes* initiate a procurement process by stating the need for a new system, to replace an existing one. The *requirements process* defines user and system requirements. The *contracts & purchasing* process uses the whole or a subset of these requirements in an invitation to tender. Tenders are evaluated and a contract is agreed between the contractor and the organization. The organization then launches a *contractor monitoring process* where technical progress is monitored. If systems, or components thereof, are safety-critical, a *safety process* has to be performed, in which a safety case is constructed that demonstrates that the system meets the organization's safety criteria. After the contractors finish development work, an *acceptance testing process* is performed in order to determine whether the system meets the user and system requirements.

An important point, which leads to serious implications for the use of software process technology, is the intertwining of different subprocesses of the overall procurement process. The subprocesses are owned and performed by, possibly competing, organizational units. Yet these units have to co-operate and the subprocesses have to be integrated in order to achieve a smooth procurement. Operational staff have to co-operate

with requirements engineers to draft the user requirements. Contractors have to cooperate with the organization's engineers during monitoring. Lawyers have to co-operate with requirements engineers in order to draft contracts. Users and requirements engineers have to co-operate with testers in order to determine acceptance.

3 Procurement Process Problems

In this section we describe the essential problems that we identified in our study. For each problem we give direct evidence that we found in the organization. These problems lead us to recommend process technology, such as the definition of standard processes, which can be tailored, and templates defining document types as an action to improve the procurement processes.

3.1 Lack of Process Standards

The processes we examined were project-driven. There was no process definition for the whole organisation. Apart from inefficient internal processes, this also caused problems during procurement. The same suppliers were contracted for different projects. They were confused because they had to adopt different processes in similar projects of the same organization. This also led to inefficiencies on the part of contractors as they could not build upon experience gained in previous projects.

3.2 Problems due to Process Autonomy

The degree of involvement of the organization in determining the contractors' processes has to be limited. On the one hand it is essential that the organization maintains sufficient control over the contracting process that it can have effective oversight. On the other hand prescribing the process restricts the choice of contractors in a commercially undesirable way and may result in inefficient processes.

The result of this tension is a rather haphazard adoption of processes and supporting technology, dependent on particular individuals. An example is a project in which a contractor used a requirements management tool that could not be accessed by the organization due to lack of a suitable hardware and operating system infrastructure. This led to higher costs and delays when traceability had to be established between the high-level requirements provided by the organization and the requirements refined by the contractor.

3.3 Lack of Process Integration

A consequence of the lack of process definitions and the process autonomy of external, and to some extent also of internal, contractors is that processes are only poorly integrated. The organization developed a user interface prototype for a new system. The development involved numerous engineers and domain experts and was performed over a period of 18 months. After finishing the prototype, it was shown to the contractor. The contractor sent a team of junior developers to view the prototype and did not take lessons

learned from the prototype into account when building the system. The reason for that was that the prototype was not tied into the contracts and procurement process. The supplier simply had no contractual obligation to take the prototype into account.

4 Impact on the Process Research Agenda

It is not sufficient to provide process support that considers software engineering processes as stand-alone and isolated processes, performed by a homogeneous project team. Rather, software processes have to be considered as embedded into procurement, system engineering or other business processes. They consist of numerous autonomous processes that are performed by different, competing, and possibly geographically dispersed units of the organization or external contractors. Software process technology has to take this into account in order to provide reasonable process support.

4.1 Need for Process Interoperability

The procurement processes we looked at consisted of numerous autonomous subprocesses. These subprocesses have to be inter-operable, even in the presence of autonomy and heterogeneity. There is need for inter-organizational process integration. The subprocesses have to be considered as independent in the sense that they are performed by different departments with different goals but they have to be glued together into one procurement process. Process integration has to take place across organizational boundaries of European partners. The processes of these organizations have to be integrated in order to have a coherent procurement and systems engineering process.

In terms of software process technology, the integration has to cover the process modelling as well as the process enactment. Process models have to be integrated into one coherent process description. For this purpose a notion of collaboration has to be introduced into process modelling formalisms. During enactment different process-sensitive software engineering environments have to be able to communicate in order to integrate autonomous processes. It cannot be assumed that all departments or organizations participating in one process are using the same modelling formalisms and runtime supporting environments. Hence PSEEs have to be open and inter-operable. Our particular interest in this area is in distribution middleware that support the construction of inter operable PSEEs [4].

4.2 Extend Scope to Procurement Organizations

Software is not necessarily built in-house but can be procured from external (or internal) suppliers. Software engineering standards, such as ISO-12207 [5] have recognised this and include practices on the collaboration between procurer and contractor. The process control has to be handed over to the contractor when the contract is signed. It is taken back from the contractor for acceptance test. Hence the system and software engineering process of the contractor is threaded into processes of the procurer. While the contractor is performing its development process, the procurer can run an independent contractor monitoring process. In this case, the coupling between different processes is not as tight as in the integration case described above.

4.3 Organizational Learning across Projects

Unlike software processes where project teams tend to be stable for a longer period of time, staffing in procurement processes is extremely fluid. Staff are brought into project teams when particular skills, such as contracting techniques, domain-knowledge or system engineering experience, are needed. It is very difficult to achieve organizational learning [2] in a project-based culture. Yet these cultures could benefit significantly as organizational learning facilitates reuse of experiences, standards and processes. We are investigating whether techniques for implementing an organizational memory for software development [3] are applicable to procurement processes.

References

1. M. Ryan A. Finkelstein and G. Spanoudakis. Software Package Requirements and Procurement. In *Proc. of the* 8^{th} *Int. Workshop on Software Specification and Design, Schloss Velen, Germany*, pages 141–146. IEEE Computer Society Press, 1996.
2. C. Argyris and D. A. Schoen. *Organisational Learning II – Theory, Method and Practice*. Addison Wesley, 1996.
3. E. Ellmer and D. Merkl. Considerations for an Organisational Memory for Software Development. In *Proc. of the* 10^{th} *Int. Software Process Workshop, Ventron, France*. IEEE Computer Society Press, 1996.
4. W. Emmerich. CORBA and ODBMSs in Viewpoint Development Environment Architectures. In M. Orlowska and R. Zicari, editors, *Proc. of* 4^{th} *Int. Conf. on Object-Oriented Information Systems, Brisbane, Australia*, pages 347–360. Springer, 1997.
5. ISO/IEC. *International Standard, Information Technology Software Life Cycle Process. ISO 12207*, 1995.

This article was processed using the LaTeX macro package with LLNCS style