Abstract

There are several initiatives for quality control in the supply of software and related services. In general, the most known and applied are the result of initiatives on the part of the customer: well defined acquisition processes, supplier selection based on certification or maturity assessment, contract monitoring. All these initiatives are aimed either to give confidence about the supplier capabilities or to directly control the development activities during their enactment.

On the side of the supplier, efficiency in the development of the software product is the most important issue, as it is the mean to be profitable. Evaluating the performance of a project has for the supplier the same importance that controlling the quality of the product has for the customer.

This paper presents a method for evaluating the performance of a software development project. The method implements a new approach that gives the supplier the ability to gather data about the performance of its projects while complying with the control initiatives of the customer.

Keywords

software acquisition, software process assessment, contract monitoring, project performance
1 Introduction

Without a well defined, enacted, and controlled development process it is almost impossible to efficiently deliver software quality. As product quality is the key factor for customer satisfaction and efficiency is the mean to be profitable, suppliers have strong and motivated arguments to control both quality and performance of their development projects.

On the customers side, the main concern is to receive in due time the required software system. Customers are not strictly involved in the economical aspects of the supplier’s process: as far as the development process of the supplier is not related to product quality, the customer is not interested in its details, neither in the causes that ended in a delay or in low product quality. Customers want to play an active role in quality control, but their scope is limited to gaining confidence about the capabilities and the controllability of the supplier process during the enactment of a contract.

This commitment on the part of customers has been the motivation of several initiatives. Some of them are aimed to prevent risks, either by supporting the customer in the choice of the “best” supplier or by allowing the customer to manage a mature acquisition process. Software Acquisition (SA) methodologies directly aim to provide guidelines and tools to organize and control the customer acquisition process. The success of Software Process Assessment (SPA) methodologies is largely due to the “certification” they provide about the capabilities of the assessed software company. Customers can use this certification in the selection phase, both for identifying the “best” supplier and for verifying the controllability of its development process.

There are other initiatives that support the customer during the project enactment for strictly and directly controlling the supplier activities. An example of this kind of initiatives is Contract Monitoring (CM), as defined by acquisition methodologies or, as a particular case, by the rules that, by law, are applied to Information Technology (IT) contracts of Italian public bodies. Even in this case the initiative is on the part of the customer.

In order to comply with the control initiatives of their customers, suppliers must dedicate some noticeable effort to demonstrate their capabilities with SPA and to support the requirements of the CM control activities.

This paper presents a method for Software Project Performance Assessment (SPPA). SPPA is a new approach in the control of software projects. Presented as an initiative of the supplier, SPPA aims at capitalizing the effort spent by the supplier to comply with the control initiatives of the customer. The idea of SPPA is to build up a set of activities, performed by the supplier, whose results:

- can be presented to the customer to fulfill its need of confidence in the capabilities of the supplier and in the correct management and progress of a specific project;
- can be used by the supplier to assess its performance during the project, providing means to evaluate the efficiency of its development process.

Section 2 of the paper presents the general context of the current control initiatives, briefly describing those from the part of the customer. Section 3 introduces the general goals of SPPA. Section 4 describes the details of SP3E, our proposed method for SPPA. Section 5 presents some examples from the first experimentations with the SP3E method. Conclusions and future work are in Section 5.

2 Control initiatives on the part of the customer

In this section we give an overview of three control initiatives that are on the part of the customer: software acquisition process, software process assessment, and contract monitoring.
2.1 Software acquisition process

One of the most critical activities performed by large organizations is the acquisition of goods and services. The acquisition of software products (SA in short) is one of the many instances of this process and it is now considered as part of the software process. The ISO/IEC 12207 [1] standard includes SA as the initial Primary process, the one that starts the software life-cycle. In the same way, the software process model defined in SPICE [2], and now part of the ISO/IEC TR 15504 [3] standard proposal, identifies SA as the first process listed in the Customer-Supplier category.

The Capability Maturity Model (CMM) [4] principles were applied to the SA process itself. The result is SA-CMM [5], an assessment methodology that can be used to evaluate or improve the SA process of an organization, or as a set of guidelines that can help in the definition of a mature SA process.

In Europe, since the mid-eighties the SA process was considered of crucial importance. The European Commission sponsored the Public Procurement Group (PPG) to define the technical rules for the acquisition of IT product and services. The most important project carried out by PPG is Euromethod [6], aimed to investigate on the relationship between customers and suppliers, to define a set of common concepts, and to find an agreement on a common terminology.

2.2 Software process assessment

Process assessment is the evaluation, performed according to a well-defined framework, of the development process of a software company. The term Software Process Assessment and Improvement puts together the goals that drive evaluation: definition of the process quality level, identification of the improvement strategies.

The most widely known SPA model, the already cited CMM, was developed as a tool for classification of suppliers. In the CMM perspective, the evolution of the software process is a path of five steps that identifies the maturity of a software company and the actions needed to move from one level to the higher one. In this sense, not only CMM is a classification tool but also a concrete methodology for process improvement.

The ISO 9000 series of standards is a commonly accepted reference that, in some market areas, has been elevated to the rank of mandatory requirement for having access to bidding procedures. Although ISO 9001 [7] is a general series of standards that address the quality system instead of the software development process, it shares many aspects with SPA [8]. For instance, the Bootstrap project [9] merged in a single SPA methodology the CMM approach and the ISO 9001 requirements for quality system management.

The SPICE (Software Process Improvement and Capability dEtermination) project aimed to establish a framework in which different assessment techniques can express process capabilities with respect to a common scale and to common assessment criteria. With ISO/IEC as a partner in SPICE, the development of project results is currently published as a ISO/IEC TR 15504.

2.3 Contract monitoring

Monitoring (CM in short) is a quality control activity performed during the enactment of a contract. As part of the SA process, CM is defined in several contexts. ISO/IEC TR 15504 defines “CUS 1.3 Supplier monitoring” as a component of the “CUS 1 Acquisition” process. In ISO/IEC 12207 monitoring is an activity of the SA, listed as a primary process. Monitoring the supplier during the contract is an activity defined by Euromethod too. ISO 9001 standard requires the Quality Plans, as documents that support the monitoring activities of the customer.

In Italy, SA procedures to be followed by public bodies are ruled by specific laws that prescribe CM and describe this activity in details [10, 11]. The Authority for Information Technology in the Public Bodies (AIPA, using its original acronym for “Autorità per l’Informatica nella Pubblica Amministrazione” [12]) defined CM with respect to both methodological and normative issues.
According to the AIPA definition, the aim of CM is to control specific facets of the contract enactment, which, though bound to the quality of the final product, are not directly related to the phases of the development process. Four “specialized” facets of CM have been identified: supplier process monitoring, project management monitoring, product quality monitoring, investment benefit monitoring.

Acceptance test is the traditional control that the customer applies to the product when it is ready to be delivered. The acceptance test, however, cannot face all the risks the customer is exposed to: it cannot face delays in the delivery, and, even worse, when it discovers defects in the product quality, acceptance test introduces further delays for their removal. The cost of these delays, even in terms of lost revenues, can overcome the whole cost of the software system. In this perspective, CM represents a customer initiative that neither replaces nor overlaps with acceptance test, but aims to directly control the supplier during the development of a required product or the provision of a service. CM represents the will of the customer to actively participate to the development process of a required and crucial product [13].

3 Software Project Performance Assessment

The importance of fully controlling the acquisition of software products and services is becoming more and more evident to customers. Clear signs of this trend are the initiatives aimed at defining, controlling, and monitoring the acquisition process. In particular, contract monitoring is an even stronger element that gives evidence of an active presence “inside” the supplier development process.

According to the 12207 standard, the software acquisition process is a starting point of the whole lifecycle. The introduction of CM as a relevant component of SA makes obsolete this position: SA is no more that runs before but a process that runs in parallel with respect to the development process. For the manager of a software development process, SA has become the reference counterpart.

The clever supplier gives paramount importance to the wishes of its customers. Following this simple and widely accepted paradigm, software suppliers must adapt to and fulfill the customer requests. In our particular context, SPA procedures are a convenient strategy that suppliers can follow to give evidence of their reached maturity level and of their capability of satisfying the commitment prescribed by contractual bindings. In order to give evidence of the capabilities of properly conducting projects and of fulfilling the deadlines contractually stated, suppliers must also accept the strict control regime imposed by CM.

Customer maturity in general requires higher costs for the suppliers. These costs are, however, balanced by the implicit advantages that the supplier can obtain from these additional constraints. The primary interest for the supplier is to improve the process efficiency, then efficiency can be exploited to obtain higher marginal financial benefits, and/or to release precious resources.

Software project performance assessment is an initiative that gives to the supplier the ability to observe and evaluate the performance of a project and, hence, the efficiency effectively exhibited during their enactment.

4 The SP³E method

SP³E, which stands for Software Process and Project Performance Evaluation, is a method for the performance assessment of software projects. The SP³E assessment technique is partially derived from the rapid assessment methodology developed in the ESPRIT/TOPS project [14].

In SP³E performance is defined in terms of efficiency: the SP³E method measures it with respect to costs and product dimensions. Other parameters can be considered part of a performance definition, for instance the quality of the product. But, in a supply context, we can assume that the product will eventually pass the acceptance test. In this case, since the quality goal is accomplished it is more important to evaluate its costs, for instance because the first acceptance test has failed and the supplier must put extra effort in debugging the software. In other words, bad product quality is the cause of poor overall performance and not another kind of poor performance.
4.1 SP³E activities

From a practical point of view, the application of the SP³E method on a given software project consists in three distinct activities: process assessment, project monitoring, and performance evaluation.

**Process Assessment.** This activity is performed at the very beginning of the software project. Its general goals are to acquire a detailed knowledge of the development process and to assure its controllability. To evaluate performance we need to keep track of schedule, product dimension and costs. Technically speaking, this means that management activities and some development activities have to be carried out at level 4. It is important to note that, during the process assessment, it is possible to discover some mandatory improvements. The assessment result is the certification that level 4 is in place, or the identification of the needed changes to bring the process at level 4. The practical result of the assessment is the “map” of the process adopted in the project. This map will be used as a guide to monitor the project activities. The assessment performed is compliant to ISO/IEC TR 15504. Maturity evaluation and general improvement suggestions may be welcome “side results” of the assessment.

**Project Monitoring.** This activity is performed during project enactment. Its general goals are to keep track of the changes to the process and to measure, for each activity, the effort and the dimension of the developed products. The need of a monitoring activity is to assure that all changes to the process are tracked. Often, the process is defined and managed, but changes are not documented, especially those deriving from unexpected problems. In a performance assessment this means the presence of hidden effort. In other cases, there are changes that do not produce a performance loss, even if they have a negative impact on the project. For instance, an activity can be delayed because of resource shortage, as far as penalties are not applied, costs and revenues do not change. As an additional result, data from the project monitoring activity can be presented to the customer to provide evidence of the good management of the project.

**Performance Evaluation.** This activity is generally performed at the end of the project. It consists of the analysis of the data collected during monitoring to evaluate the overall performance of the project. Punctual performance can be evaluated also during the enactment of the project, but we have to wait until the very end of the project to have a stable configuration of data. The data can be analyzed and compared with reference models.

SP³E activities should be carried out by professionals that are independent from the supplier company. Apart of the performance evaluation, this can also help if results are presented to the customer, for instance in the context of CM. SP³E activities are supported by forms that help in collecting data about the project:

- **project form**, collects data about the project, the people involved, the available documentation, and the product to be delivered;
- **work-package form**; collects data about a single work-package: description, schedule, effort estimated and actual; the first configuration of the set of work-package forms is generated during the process assessment and is constantly updated during the project monitoring activity; process attributes of the single work-package are verified during the process assessment to assure the controllability of the work package;
- **component form**; collects data about a single software component: description, related work-packages, dimension; the set of component forms is generated and constantly updated during the project monitoring activity.

The current version of templates for SP³E forms and the guidelines for their compilation are available at [15]. As sketched in fig. 1, SP³E activities are organized in a way that fruitfully exploits activities that the supplier must anyhow perform in order to satisfy the new requests of the customer, in particular the CM activities performed inside the SA process.
Fig. 1. Relations between SP3E activities and customer control activities

4.2 SP3E results

The result of SP3E activities can be exploited to derive two kinds of performance evaluation of the assessed project: quantitative performance evaluation, as a direct measure of the project productivity, and qualitative performance evaluation, as a set of considerations that can help the project management to identify improvements suggestion for their development process.

Quantitative Performance Evaluation. In SP3E we focus on efficiency as the most important facet of project performance. Efficiency can be measured as a ratio between product size and effort or between product size and costs. In SP3E product size is measured using Object Points as defined in the proposal of the CoCoMo 2 model [16]. Effort is measured in man/month, as usual. The size/effort measure, however, is a rough approximation of the efficiency because there are substantial differences among man/month costs of differently skilled personnel. For this reason SP3E measures the industrial cost of each activities and produces the size/cost ratio as a measure of the efficiency.

Qualitative Performance Evaluation. Apart of performance measures, SP3E provides several “maps” that can be useful in the study of the project performance. Basically, SP3E maps are coloured Gantt charts. We use Gantt charts as basic representations of project activities. Then we use different colours to highlight activities that differ with respect to a characteristic that influences performance. For each characteristic SP3E provides a map, the different colours describe how the activities are differently positioned with respect to that characteristic. At the moment we have identified three characteristics: effort cost, to highlight cheap and expensive activities, size/time, to highlight the activities with respect of time to deliver, tools dependence, to highlight the activities with respect to the different development tools used for their enactment.

Quantitative performance evaluation is proposed mainly as a benchmark tool. In its first approximation, size/effort, the performance result can be compared with the "standard" value that can be obtained applying the CoCoMo 2 model to the project. The size/cost result can be compared with data
Qualitative performance maps are proposed as an investigation tool. A typical way to study the maps is to compare them and to search correlations among characteristics. For instance, it is possible to “see” if the introduction of a development tool was useful with respect to time to deliver or to cost reduction.

5 Experimentation with the method

The SP³E method is yet in development. Several trials were conducted to experiment with the method but, currently, no complete application of the method in the industry has been done. The main motivation is that, in the trials we did, the initial assessment showed too poor maturity levels. If the process is not improved, it is impossible to keep track of activities and costs to evaluate the performance. As it is too expensive to change the process, the experiment terminated after the first phase. Even if this is quite frustrating for us, it is indeed a result for the company. Performance evaluation seems to be a good lever for convincing companies to perform an initial process assessment. The proof of evidence that they are not able to have a reliable assessment of their performance is a strong motivation to start an improvement path.

Having failed our first attempts to experiment the method in an industrial setting we directed our efforts to experiment the method with students involved in software engineering projects. In the course of Tecniche di Programmazione (Programming Techniques, TP), held at the University of Florence, students are grouped in development teams that act like small software companies. The goal of the course, that in the curriculum follows a classic Software Engineering course, is to apply in a practical project both development techniques (analysis and design using UML, testing) and management techniques (cost estimation, project planning and control).

In the simulation proposed by the course there is a customer that asks for a software application. Teams must perform requirement analysis, present a formal offer that includes a price, and deliver the application in the given time. An important constraint imposed to teams is to stay in the cost estimation from which they derived the offer price. This year there were six teams, grouped in three consortiums. Teams were required to define their process and to consistently plan and control it. To assure that cost are estimated, activity planned and controlled, delays tracked and cost accounted, teams were subjected to three formal inspections, whose results influenced the final score for the course. Before each inspection, teams were required to consolidate activities and costs and, if necessary, to update their project plans.

In the project, according to the performed activities and their declared skills, students can play several roles and each role has a different cost. For instance, a student who already had followed a specialized course in Java was considered a senior Java programmer, while a student who had only followed a general programming course was considered a junior programmer. The same rules applies to other skills participating to the project (Java, SQL and UML) and to activities that involve more responsibility (such as analysis, design, and management). The student teams had to manage these issues as if they were in a true company. Their plans were required to keep track of all the changes about duration effort and responsibility (as in a “mature” company).

From the performance assessment point of view, the student teams, while not a real software company, acted perfectly because their process was well defined and data were consistently kept. In this perspective their projects were subjected to a strict monitoring activity and, hence, well suitable for experimenting the SP³E method. In the following we show an example of an SP³E map drawn using the data of two student teams and we use these maps to discuss their performance.

Figure 2 shows two effort cost maps. They present the high level activities of two teams (analysis and offer proposal, design and coding, integration and verification) and indicate how the average cost of activities changed during the project. The upper bar represents initial planning, the lower bar represents actual results. The length of the bars is proportional to effort (not time of deliver but how many man/hours were totally consumed). The number inside the bars is the average cost per hour (the same information is given visually by the bar colour: light means cheap, dark expensive).
Using the maps of Figure 2 it is possible to evaluate the performance of the two teams with respect to role management. Both teams faced an increased effort with respect to initial planning. Team A was very efficient in managing the extra required effort without too much increasing the overall costs. In fact, the average cost per hour decreased. Going into more details (using the tracked management data) we notice that, for instance, in the offer phase, when they realized that the number of meetings with the customer was rapidly increasing they decided to use lower skilled people for collecting information and for writing the initial version of the analysis documents, saving the time of skilled and more expensive people for the last revision of the documents. A similar strategy was used for testing, where, to stay in the budget, they made the most relevant cut deciding for a naïve verification rather than designed test suites when they were more confident that a component was well coded. Even if the purpose of the method is not to compare, but rather to investigate and discuss the efficiency of a project, in this example it is clear that Team A exhibits a more efficient resource management.

6 Conclusions and future work

In this paper we identify a new activity in the context of quality control of software project: SPPA. This activity is organized to fruitfully exploit the effort that the supplier must anyhow perform in order to satisfy other control initiatives requested by the customer: SPA and CM. We also propose a method, SP³E, for performing SPPA. The method is briefly presented in general and example of its use is discussed as a mean to evaluate the performance of two development teams with respect to role management.

Our method is focused on efficiency, as the main indicator of performance. However, efficiency is not the only performance indicator that we deem important for SPPA. Other indicators, when combined with efficiency, can give a more precise and complete picture of the performance of a software devel-
opment project. For instance, we have in mind to use the competence level of the personnel associated to software projects and the capability of compressing activities along the time dimension, both by using at best the available technological tools and by increasing the automation level of the development process. We are fully aware that the research is still at an initial stage but we actively plan to validate the SP3E method by means of experimentation on real industrial projects.

The definition of a method for evaluating software project performance was partially supported by the Estate project [15], funded in the context of the Lyee International Collaborative Research Project. Lyee [17] is a software development methodology based on a specific analysis method and a tool for automatic code generation. In the Estate project the SP3E method will be used to assess the performance of the projects carried out using Lyee.

7 Acknowledgements

The affiliation of G.A. Cignoni is with the University of Pisa, because the research about SPPA was developed there, but the reported data came from a course held by him at the University of Florence. We want here to tanks the students of the course of Tecniche di Programmazione.

8 Literature

Some of the listed references cite Italian laws: [10] is the Italian law where monitoring was defined for the first time; [11] is the first edition of the official guidelines for monitoring.

9 Author CVs

Vincenzo Ambriola

Vincenzo Ambriola is associate professor of Software Engineering at the Department of Computer Science of the University of Pisa. Author of more than fifty scientific publications on international journals and conferences, he is currently in the editorial board of Software Process Improvement and Practice. Vincenzo Ambriola has been member of the program committee of more than twenty international conferences on Software Engineering (European Conference on Software Engineering, Software Engineering and Knowledge Engineering, Software Maintenance and Reengineering, among the others). He has been principal investigator of numerous research projects sponsored by the Italian industry as well as the National Council of research. His main research interests are in Software Engineering and, in particular, in the area of Requirements Engineering and Software Design.

Giovanni A Cignoni

Giovanni A. Cignoni received in 1992 the Laurea degree in Computer Science from the University of Pisa. From 1993 he is a freelance consultant in Information Technology, with particular expertise in the fields of software engineering, software quality, and software process management. As consultant of the Dept. of Computer Science and of the Dept. of Mathematics of the University of Pisa, he participated to Italian and European research projects. His main research interests are in Software Engineering and in particular in the area of management and control of the software development process. As contract professor he held courses about software engineering at the Universities of Pisa, Florence and Padova. He is author of international publications and co-author of the books “Laboratorio di programmazione” (in Italian, “Programming workbench”) and “Il test e la qualità del software” (in Italian, “Test and software quality”). 
